

Monitoring the High Energy Universe

Open, Reproducible, Machine Learning Based Analysis
for the First G-APD Cherenkov Telescope

Maximilian Nöthe
2020

A document submitted in partial fulfillment of the requirements for the degree of
Doctor rerum naturalium
at
Fakultät Physik, Technische Universität Dortmund

Supervised by
Prof. Dr. Dr. Wolfgang Rhode and Prof. Dr. Carsten Westphal

This thesis is set in Libertinus (Serif, Sans and Math) and Fira Code,
typeset using \LaTeX with Lua \TeX from \TeX -Live 2020.
Title graphic by Sebastian A. Müller.

Abstract

High energy gamma-ray astronomy probes the most extreme phenomena in our universe: super novae and their remnants as well as supermassive black holes at the center of far away galaxies. The First G-APD Cherenkov Telescope (FACT) is a small, prototype Imaging Air Cherenkov Telescope (IACT) operating since 2011 at the Roque de los Muchachos, La Palma, Spain. It specializes in continuously monitoring the brightest known sources of gamma rays.

In this thesis, I present a new, open analysis chain for the data recorded by FACT, with a major focus on ensuring reproducibility and relying on modern, well-tested tools with widespread adoption.

The integral sensitivity of FACT was improved by 45 % compared to previous analyses by the introduction of an improved algorithm for the reconstruction of origin of the gamma rays and many smaller improvements in the preprocessing. Sensitivity is evaluated both on simulated datasets as well as observations of the Crab Nebula, the “standard candle” of gamma-ray astronomy.

Another major advantage of this new analysis chain is the elimination of the dependence on a known point source position from the event reconstruction, thus enabling the creation of skymaps, the analysis of observations where the source position is not exactly known and sharing reconstructed events in the now standardized format for open gamma-ray astronomy. This has led to the first publication of a joined, multi-instrument analysis on open data of four currently operating Cherenkov telescopes.

A smaller second part of this thesis is concerned with enabling robotic operation of FACT, which is now the first Cherenkov telescope, where no operators are required during regular observations.

Zusammenfassung

Die Hochenergie-Gammaastronomie erlaubt es, die extremsten Phänomene in unserem Universum zu untersuchen: Supernovae und ihre Überreste sowie supermassive schwarze Löcher in den Zentren weit entfernter Galaxien. Das First G-APD Cherenkov Telescope (FACT) ist ein kleines, bildgebendes, atmosphärisches Tscherenkow Teleskop, das seit Oktober 2011 auf dem Roque de los Muchachos, La Palma, Spanien beobachtet. Es ist auf die Langzeitbeobachtung der hellsten bekannten Gammastrahlungsquellen spezialisiert.

In dieser Arbeit stelle ich eine neue, öffentliche Analyseketten für die von FACT aufgenommenen Daten vor. Ein Hauptaugenmerk wurde auf die Reproduzierbarkeit und die Verwendung moderner, gut getesteter und weit verbreiteter Methoden gelegt. Die integrale Sensitivität von FACT wurde im Vergleich zu früheren Analysen um 45 % gesteigert, hauptsächlich durch die Einführung einer verbesserten Methode zur Bestimmung der Herkunft der Gammastrahlung, sowie durch viele weitere, kleinere Verbesserungen in der Vorverarbeitung. Die Sensitivität wurde sowohl auf simulierten Daten als auch auf Beobachtungen des Krebsnebels, der Standardkerze der Gammaastronomie, ausgewertet.

Ein weiterer Vorteil der neuentwickelten Analyseketten ist ihre Unabhängigkeit von Annahmen über eine bekannte Punktquelle. Dies ermöglicht die Erstellung von Himmelskarten, die Analyse von Beobachtungen, bei denen die Quellposition nicht genau bekannt ist und das Speichern und Veröffentlichen rekonstruierter Ereignisse im nun standardisiertem Datenformat für offene Gammaastronomie. Dies hat die Publikation der ersten gemeinsamen Analyse von Krebsnebel-Daten von vier aktuell beobachtenden Tscherenkow-Teleskopen ermöglicht.

Der zweite, kleinere Teil dieser Arbeit beschäftigt sich mit der Robotisierung von FACT, welches nun das erste Tscherenkow-Teleskop ist, für dessen reguläre Observationen kein Personal mehr benötigt wird.

Contents

I	Sensitivity of the First G-APD Cherenkov Telescope	1
1	Gamma-Ray Astronomy in the Context of Astroparticle Physics	3
1.1	Cosmic Messengers	4
1.2	Observing Cosmic Gamma Radiation	5
1.3	Sources of Gamma Radiation	9
1.4	Key Science Questions	14
1.5	Cosmic Rays	14
2	The First G-APD Cherenkov Telescope	17
2.1	Photosensors and Readout Electronics	18
2.2	Observation Strategy	21
3	Monte Carlo Simulation of FACT Events	23
3.1	Simulation of Air Showers Using CORSIKA	23
3.2	Simulation of Detector Response using CERES	26
3.3	Reweighting Simulated Events to Physical Fluxes	27
3.4	Monte Carlo Production Software	27
3.5	Comparison of FLUKA and URQMD	29
4	From Raw Data to Image Parameters	31
4.1	Data Format and Structure	31
4.2	FACT-Tools	32
4.3	Raw Data Calibration	32
4.4	Signal Extraction	33
4.5	Image Cleaning	36
4.6	Image Parameterization	36
4.7	Output file format	38
4.8	Celestial Coordinate Systems and Transformations	38
4.9	Processing of all FACT data to Image Parameter Level	42
5	Reconstruction of Particle Properties using Machine Learning	45
5.1	Supervised Machine Learning	46
5.2	Particle Classification	51
5.3	Energy Estimation	51
5.4	Origin Estimation	51

6 Instrument Response Functions and Sensitivity of FACT	55
6.1 Datasets	55
6.2 Detection Significance of the Crab Nebula	63
6.3 Separation Performance	66
6.4 Instrument Response Functions	69
6.5 Differential Flux Sensitivity	76
7 Energy Spectrum of the Crab Nebula	81
7.1 Inverse Problems	81
7.2 Poisson-Maximum-Likelihood Unfolding	82
II Automatization of the First G-APD Cherenkov Telescope	85
8 Robotic Operation of FACT	87
8.1 Remote Control Infrastructure	87
8.2 The shiftheelper	88
8.3 Webinterface	91
8.4 Heartbeat	92
9 Conclusions and Outlook	93
Bibliography	95
Glossary	109
Appendix	113
Affidavit (Eidesstattliche Versicherung)	139
Acknowledgements	140

Part I

Sensitivity of the First G-APD Cherenkov Telescope

Gamma-Ray Astronomy in the Context of Astroparticle Physics

1

The last two decades have been revolutionary for astronomy. For the first time, observations are routinely carried out over the whole electromagnetic spectrum from radio wavelengths to very high energy gamma radiation. Joining in, first observations of gravitational waves in 2015 [5] and a first hint of neutrinos [2] from a gamma-ray source bring us to the verge of true multi-messenger astronomy. Only provided by the full picture of the universe as delivered by photons over the whole electromagnetic spectrum, neutrinos and gravitational waves, can we hope to understand the processes governing the high energy universe, which produces particles at energies out of reach for particle accelerators on Earth. The field of astroparticle physics combines techniques, methods and theory from high energy particle physics to study the universe through these high energy cosmic messengers reaching Earth.

In the following sections, I will give an introduction into the general field of astroparticle physics in [section 1.1](#), especially gamma-ray astronomy, and the crucial role it plays in understanding our universe. I will present the currently operating and planned gamma-ray observatories in [section 1.2](#), the most common sources of gamma radiation in [section 1.3](#) and the key questions that are currently pursued in [section 1.4](#). I will discuss charged cosmic rays, mainly focussing on the role they play in gamma-ray astronomy in [section 1.5](#).

After the general introduction into the field, I will introduce the First G-APD Cherenkov Telescope (FACT), the instrument used to perform my studies, in [chapter 2](#), its mission and properties important for the context of my work.

Due to the non-existence of artificial sources for calibration, high-energy astrophysics experiments must heavily rely on simulations. This is necessary for tuning of detector parameters, estimating performance and creating datasets with known physical properties to be able to reconstruct these properties for recorded events. The simulations performed for this work are detailed in [chapter 3](#).

[Chapter 4](#) will present the data analysis techniques used to make sense of the raw data and how the vast amounts of data are reduced to a more manageable size while keeping most of the relevant information via the use of few but descriptive properties.

From these features, estimators for the physical properties of the detected particles are created through the means of statistical learning. This is the topic of [chapter 5](#).

The first part of this thesis is concluded with the presentation of the sensitivity and instrument response functions of the combined system of telescope and data analysis in [chapter 6](#). These will be evaluated both on simulated and observed data where possible. Finally, estimating the energy spectrum of the Crab Nebula will be the topic of [chapter 7](#).

In a smaller second part, the effort put in to make FACT a robotically operating telescope, observing on its own nearly without the need of human intervention or supervision, is presented in [chapter 8](#).

At the very end, synopsis and prospects of further work is provided in [chapter 9](#).

1.1 Cosmic Messengers

Information about the universe is reaching Earth through multiple kinds of radiation and stable particles. Each kind has its own challenges in detecting the messengers and extracting the contained information. The different messengers are also unique in the kind of information they are able to convey.

Since the dawn of humanity, people have looked into the sky and observed it in the visible spectrum, first with naked eyes then with ever larger and better telescopes. Radio observatories enabled a first glimpse at the non-thermal universe and observing the thermal remnants of the Big Bang. The advent of space flight made telescopes not hindered by absorption of light in Earth's atmosphere, light pollution and adverse weather conditions possible. Together with photographic plates and electronic photodetectors, this opened up observations in wavelengths not accessible before, including the far infrared, X-rays and gamma rays. Finally, the Imaging Air Cherenkov Telescope (IACT) technique allows for observations of gamma ray sources at energies larger than 100 GeV. Observations of gamma rays will be discussed in detail in [section 1.2](#).

Earth is constantly bombarded by high energy charged particles, cosmic rays. These particles reach energies of up to 10^{20} eV, which are far out of reach for today's particle accelerators. The question of their origin is still largely unanswered and is connected to the question of the acceleration mechanisms at work. Unfortunately, due to deflection in interstellar and intergalactic magnetic fields, their arrival directions at Earth do not point back to their sources at all but the highest energies. Cosmic rays are mainly of interest in gamma-ray astronomy because they form the main background for all experiments and more details on this are given in [section 1.5](#).

Neutrinos are electrically neutral and only interact via the weak force resulting in very small cross sections. While this makes them the ideal messenger particles for many processes as they can escape fast even from dense regions of interest, are not deflected in magnetic fields and do not suffer from absorption in the interstellar or intergalactic medium, it also makes it extraordinarily challenging to detect them on Earth. The IceCube neutrino observatory [[12](#)] instrumented one cubic kilometer of ice below the South Pole to detect neutrinos via the secondary particles they produce when they interact with the ice. In 2013, the IceCube collaboration announced the first observations of high energy neutrinos from outside our solar system [[1](#)].

The last window into the universe was only opened four years ago, when the LASER Interferometer Gravitational-Wave Observatory (LIGO) detected the first gravitational wave that was caused by two merging black holes [[5](#)], a hundred years after the existence of such waves was first predicted by General Relativity. In the mean time, LIGO and the VIRGO observatory detected more than 20 gravitational wave events, among them a binary neutron star merger that was also observed in the electromagnetic spectrum by many telescopes [[6](#)]. Gravitational waves currently observable on Earth are only created by the most extreme phenomena, merging compact objects like black holes and neutron stars.

1.2 Observing Cosmic Gamma Radiation

Currently, there are three main approaches to observing gamma radiation, each with their characteristic energy range, sensitivity, field of view and duty cycle. As the atmosphere is completely opaque to photons at these energies, direct observations are only possible in space.

At energies below the tens of GeV, space based telescopes directly measuring the gamma rays using pair production particle detectors currently have the best sensitivity. They offer a field of view of up to 3 sr and can thus monitor a large proportion of the sky simultaneously. Only a single photon per day and square meter reaches Earth from one of the brightest sources of high energy gamma radiation, the Crab Nebula, above an energy of 40 GeV¹. The small number of detectable events because of insufficient detection area is the limiting factor for satellite based telescopes [56].

Above these energies, ground-based telescopes can cover very large areas by observing the gamma rays indirectly using secondary particles created in Extensive Air Showers (EASs). Two different methods are used in existing observatories: IACTs measure the Cherenkov light produced by the charged secondary particles using optical telescopes with cameras able to observe the dim, nanosecond-long flashes of light. These telescopes have very large effective collection areas, up to several hundred thousand square meters, as the diameter of the Cherenkov light pool on the ground reaches 250 m. IACTs can only observe during dark nights and under favorable weather conditions severely limiting their duty cycle and they have a limited field of view, usually allowing only a single source to be observed at a time.

A third possibility is to measure high energy particles created in the air showers reaching the ground using large water tanks, also observing the Cherenkov radiation created by the particles, but now in the water of the tanks and not in the atmosphere. Water Cherenkov detectors have higher energy thresholds and lower point source sensitivity compared with IACTs, but can observe the whole sky above them day and night reaching duty cycles close to 100 %.

1.2.1 Direct Observation using Satellites

The most sensitive, space based, currently operating observatory is the *Fermi* Gamma-Ray Space Telescope, which is operating in low Earth orbit since 2008. Equipped with two different scientific instruments, *Fermi* is sensitive to energies between 20 MeV and 300 GeV using its Large Area Telescope (LAT), which detects photons via pair production inside the detector which covers 20 % of the sky. The LAT is a miniature version of the particle detectors used at accelerators on Earth and comprises three main components: an outer plastic scintillator shell for suppression of the cosmic ray background, a silicon strip based particle tracker for tracking the produced electron-positron pair and a calorimeter for measuring its energy.

The *Fermi*-LAT collaboration has published multiple catalogs of gamma-ray sources. The latest version, called 4FGL [8], accumulates over 7.5 years of data and lists over 5000 individual sources for energies above 50 MeV. Another catalog especially for high energy sources detected above 10 GeV, the 3FHL [17], lists 1556 sources using seven years of data. All the gamma-ray sources from the 4FGL are shown in Figure 1.4. The LAT Collaboration also performs real time analysis

¹Obtained by integrating the log-parabola parameterization of the Crab Nebula flux published by the MAGIC collaboration in [22] between 40 GeV and 200 TeV

of the data, alerting the community of possible flares, sudden increases in the flux of gamma-ray sources.

The second instrument, the Gamma-Ray Burst Monitor detects gamma-ray bursts (GRBs) between 8 keV and 30 MeV while observing the whole sky that is not shadowed by Earth, also providing fast alerts for other observatories.

Two other satellites are also important for multi-messenger efforts: AGILE was launched in 2007 and is very similar to the *Fermi* satellite but has a slightly larger field of view and a smaller collection area. It also features a GRB monitor [134]. The Neil Gehrels *Swift* Observatory is an X-ray observatory with the primary purpose of detecting GRBs operating since 2004.

1.2.2 Imaging Air Cherenkov Telescopes

Above energies around 30 GeV, ground based telescopes detecting gamma rays via the Cherenkov light produced in extensive air showers are the most sensitive instruments.

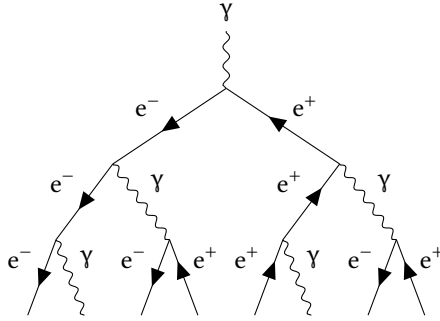


Figure 1.1: Simplified first steps of an air shower induced by a photon. In each generation, pair production and bremsstrahlung produce new particles, effectively doubling the number of particles and halving their energy. This simplified approach describes some overall parameters of purely electromagnetic showers remarkably well for such a simple model [86]. All these processes can only happen in the fields of atoms, not shown here.

When gamma rays or charged particles enter the atmosphere, they produce a cascade of secondary particles. As gamma rays and electrons do not interact via the strong force, their showers are much simpler to describe and are dominated by only two processes. Gamma rays produce electron/positron pairs in the field of molecules and electrons in turn produce new gamma rays via bremsstrahlung. This continues until the energies of the particles fall under the threshold energies for the corresponding processes, e. g. 1022 keV for pair production.

A simplified model after Heitler [86] of this is shown in Figure 1.1. The produced charged particles are faster than the speed of light in the atmosphere and thus induce the emission of Cherenkov Light. Cherenkov radiation is emitted in a fixed angle θ depending on the refractive index n of the dielectric medium and the velocity of the particle as fraction of the

speed of light in vacuum β :

$$\cos \theta = \frac{1}{\beta n}. \quad (1.1)$$

From this, the critical velocity for Cherenkov emission follows as

$$\beta \geq 1/n. \quad (1.2)$$

For a refractive index of $n - 1 = 10^{-4}$, which corresponds to a height of ten kilometers, the threshold energy for electrons is thus

$$E_{\min} = \gamma_{\min} E_0 = \frac{m_e c_0^2}{\sqrt{1 - 1/n^2}} \approx 36 \text{ MeV}, \quad (1.3)$$

where γ is the Lorentz factor $\gamma = 1/\sqrt{1-\beta^2}$.

In air showers caused by hadronic primaries, more processes become possible due to the strong interaction, for example meson and hadron production. Charged mesons and hadrons will also contribute to the Cherenkov light production. Neutral pions mostly decay into two gamma rays or a gamma ray and an electron positron/pair, producing electromagnetic subshowers. If the charged primary is absorbed early by such a process, the resulting showers are hard to discriminate from gamma or electron induced showers. Charged pions decay into muons and neutrinos. Muons also produce Cherenkov light and, if they reach the telescope, produce ring images in the cameras of IACTs. **Figure 1.2** shows the Cherenkov light reaching the ground for three different primaries: a gamma ray, a proton and an iron nucleus.

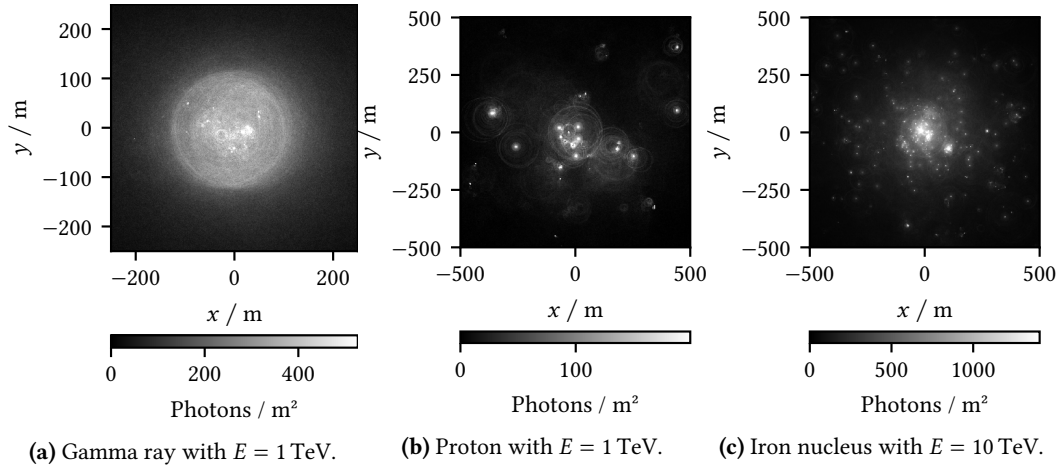


Figure 1.2: Cherenkov light reaching the ground at the telescope level for three different primary particles. Gamma rays produce rather homogeneous, circular pools of light, almost purely from electrons and positrons. Hadronic primaries produce more subshowers with high transversal momenta and produce large numbers of muons from pion decays. An illustration of the iron shower where the different components of the Cherenkov light are indicated by the color channels is shown in Appendix 1.

IACTs observe the dim flashes of Cherenkov light produced in EAS, usually with segmented reflectors and cameras comprising fast, sensitive photon detectors that can record down to single photons and have time resolutions of few nanoseconds. The physical properties of the primary particle inducing the air shower have to be reconstructed from these “videos” of Cherenkov light. Having several telescopes operating together—observing the showers from different angles—dramatically improves directional reconstruction, background suppression and collection area.

Currently, four systems of IACTs are observing regularly. The High Energy Stereoscopic System (H. E. S. S.) started observing using four telescopes with a diameter of 13 m in 2004 [91]. A fifth, 28 m telescope was finished in 2012 [91]. Located on the southern hemisphere in Namibia, one of H. E. S. S.’ primary results is the H. E. S. S. galactic plane survey [7], which provides a deep look into the TeV emissions of the Milky Way and detected 78 sources of very high energy gamma radiation.

The Major Atmospheric Gamma-Ray Imaging Cherenkov (MAGIC) telescopes are a system of two 17 m telescopes at the Observatorio del Roque de los Muchachos (ORM) on the Canary Island



Figure 1.3: The MAGIC, FACT and LST-1 telescopes (from left to right) at the ORM in April 2018, when the LST-1 was still under construction.

of La Palma, Spain. MAGIC started observations in 2004 with a single telescope, the second was added in 2009 and a major upgrade of the electronics performed in 2012 [21]. Since then, the two MAGIC telescopes are a homogeneous, stereoscopic system detecting gamma rays above 30 GeV. MAGIC is built to react quickly to alerts from other observatories, like the *Fermi*-GBM or *Swift*, being able to reach any point in the sky in just 40 seconds [21]. In January 2019, MAGIC was the first ground based telescope to directly detect TeV emission from a GRB, starting observations only 50 seconds after the alert from *Swift* was received [117]. This was one of the brightest events ever observed on the gamma-ray sky, reaching flux levels of up to 50 times the Crab Nebula flux in the first 50 seconds observed by MAGIC [10].

The four 12 m Very Energetic Radiation Imaging Telescope Array System (VERITAS) telescopes are observing from Arizona, USA, since 2007 [94]. Demonstrating what IACTs are capable of besides gamma-ray astronomy, VERITAS measured the diameter of two stars with a resolution of 0.1 mas by observing the diffraction pattern caused by an asteroid occultation of the stars [30].

The FACT is a smaller, 4 m diameter prototype telescope, with the primary goal of demonstrating the feasibility of using semi-conductor based photo detectors for gamma-ray astronomy and continuously monitoring the brightest blazars. It will be presented in detail in [chapter 2](#).

To improve upon the sensitivity of the currently operating telescopes by at least an order of magnitude, both in the northern and the southern hemisphere, a large international consortium is currently planning and building the Cherenkov Telescope Array (CTA). In its final form, it will comprise 99 telescopes at the southern observatory in Chile and 19 telescopes at the northern site at the ORM [11]. As of November 2019, prototypes for all of the three different telescope sizes are developed and the first 23 m Large Sized Telescope is in commissioning at La Palma and detected its first signal of the Crab Nebula [54]. Construction of the full array will take several years and is currently planned to be finished in 2025 [53].

As one major open question is the variability of Active Galactic Nuclei (AGN), see [subsection 1.3.1](#), dense, unbiased² monitoring of these objects is crucial to understand the underlying processes. A possible solution for this would be to install Cherenkov telescopes around the globe at multiple

²The currently observing large telescopes tend only to observe AGN in active states when triggered through an external alert, thus having a large selection bias on possible flux states.

locations so that at least one telescope can observe a source at any given time. FACT was started as a prototype for a telescope in such a ring, but having larger telescopes with higher sensitivity for the short scale variability would improve the possible scientific results dramatically [138].

1.2.3 Water Cherenkov Telescopes

The third possibility of detecting high energy gamma-radiation is measuring the energetic particles produced in air showers that reach the ground using water Cherenkov tanks. These large volume tanks contain photosensors that detect the Cherenkov light produced by particles going through the water. As these water tanks are closed, these observatories are not bound to observing only in dark nights and are much less affected by weather conditions in comparison to IACTs. Also, they observe the whole sky above them at any given time, though sensitivity is limited beyond zenith distances of more than 45° . The only currently operating observatory using this detection mechanism is the High Altitude Water Cherenkov (HAWC) experiment in Mexico [9]. HAWC is fully operational using 300 water tanks with 190 m^3 of water each since 2015, each tank is outfitted with four photo multiplier tubes (PMTs) [9]. HAWC's sensitivity for point sources is lower than for most IACTs in the common energy range, but it can monitor many sources at the same time.

1.3 Sources of Gamma Radiation

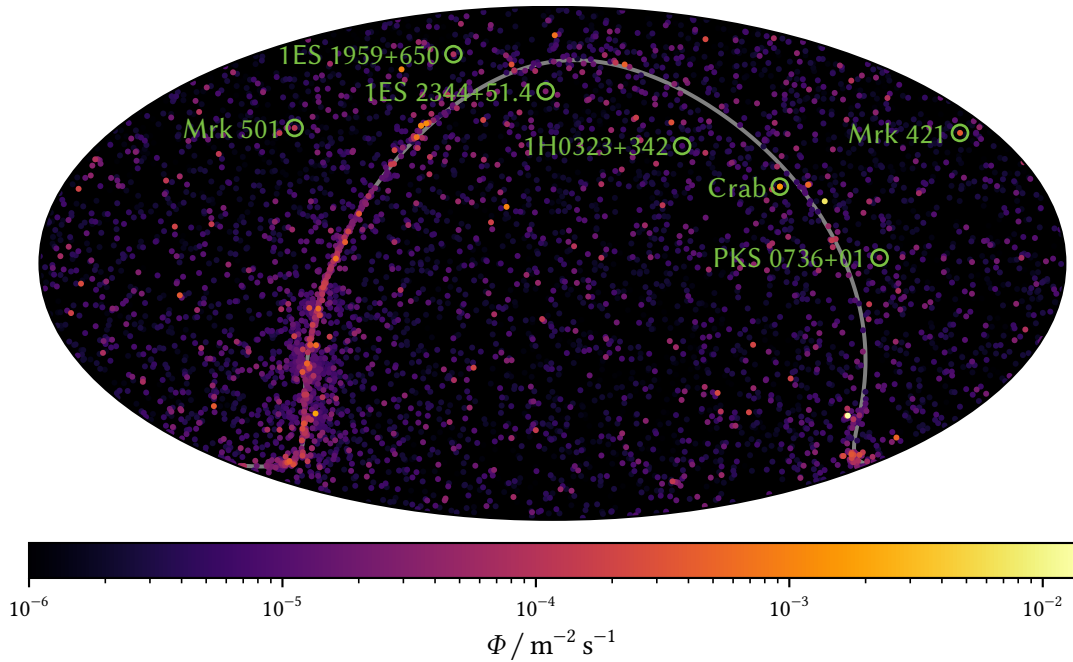


Figure 1.4: The gamma-ray sky as observed by the Large Area Telescope aboard the *Fermi*-Satellite in equatorial coordinates and equal-area Mollweide projection. The color of the points indicates particle flux between 1 GeV and 100 GeV. A gray band shows the galactic plane. The seven sources most observed by FACT are marked. All but the Crab Nebula, which is a supernova remnant (SNR), are blazars. Data taken from the 4FGL [8].

High energy gamma radiation can originate from two main source types. Gamma rays from inside the Milky Way are mostly produced in super nova remnants, mostly pulsars, Pulsar Wind Nebula (PWN) and shell-type super novae. Extragalactic sources are dominated by AGN, the very bright central regions of galaxies containing super massive black holes with 10^6 to 10^{10} solar masses M_{\odot} [57].

1.3.1 Active Galactic Nuclei

Outside of our own galaxy, the most numerous, lasting sources of gamma radiation are AGN, galaxies with super massive black holes at their centers. These black holes accrete matter from the surrounding medium, forming disks around the black hole. These objects belong to the brightest in the universe, reaching powers of 10^{40} W [57]. In some cases, narrow, relativistic jets of particles shoot out perpendicular to the accretion disk of matter falling into the black hole.

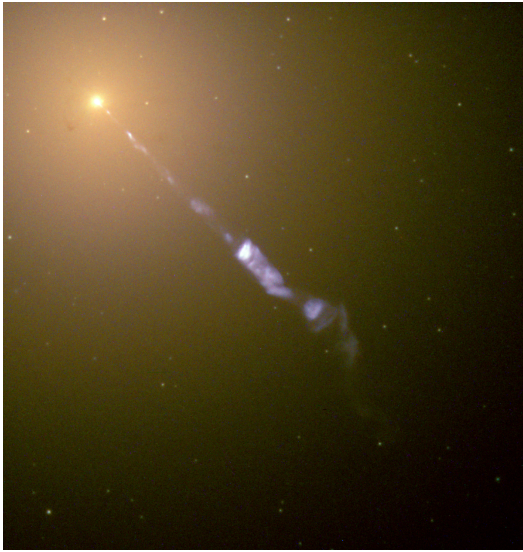


Figure 1.5: Hubble image of the radio galaxy M87, an AGN with a larger viewing angle than blazars, so the relativistic jet is visible. [118]. The central black hole of M87 was also the first to be directly imaged through Very Long Baseline Interferometry (VLBI) [18].

The jet producing AGN are classified into categories based on the viewing angle towards the jet. A viewing direction nearly down the jet and the resulting Doppler-boosting classify an AGN as blazar and these are the main sources of high energy, extra-galactic gamma rays. Blazars are subdivided into two categories based on the existence or absence of line emission in the optical spectrum. Flat Spectrum Radio Quasars (FSRQs) show broad line emissions, likely from gas clouds closely orbiting the black hole, and generally have a higher energy output than BL Lacertae objects, named after the prototypical source, that show no or very weak line emissions [146].

While BL Lac objects have lower overall energy output, they reach higher energies and are the predominant extra-galactic source type at TeV energies. The 4FGL catalog contains 1109 BL Lac objects, 652 FSRQs and 1310 blazars of uncertain type [8]. Compared to that, TeV-Cat [155], an online catalog keeping track of gamma-ray observations at TeV energies, lists 52 BL Lac objects and only 6 FSRQs to be detected in this energy range.

The gamma-ray flux of blazars is characterized over more than ten magnitudes by a double hump structure. The first, low energy hump is universally assumed to be from synchrotron radiation of high energy electrons and peaks in the infrared to soft X-ray band. The second, high energy hump is not fully understood and two main classes of models try to describe the high energy emission, either purely leptonic models or models including hadronic components.

The simplest model describing gamma-ray flux from blazars is the single zone, synchrotron self-Compton model. Assuming a single population of high energy electrons, low energy photons are

created by synchrotron radiation and some of these photons gain energy from the same population of electrons via inverse Compton scattering. This scenario is purely leptonic and would not require high energy protons, thus excluding these sources as possible origin of high energy cosmic rays.

However, some blazars show hints of a hadronic component, where additional high energy photons are created by neutral pion decays [146]. As these hadronic reactions would also produce charged pions, which in turn decay into neutrinos, observations of neutrinos from blazars would be the “smoking gun” for blazars being sources of high energy cosmic rays. A first strong hint for neutrinos from a blazar was published in 2018 by the IceCube collaboration, where first a single, high energy neutrino from a direction consistent with the blazar TXS0506+056 simultaneously with a flare in the electromagnetic spectrum of the source was observed [2]. In a second step, IceCube found an excess of lower energy neutrinos from the direction of the source in historic data with a significance of 3.5σ [3].

Blazars show large variability of their flux states, both on long time scales slowly changing their mean power over months and years and on very short time scales, doubling their flux in mere minutes, contesting the more simple models for blazar emission that cannot describe variability on such short timescales [146]. In a very bright outburst in 2005, MAGIC observed a doubling of the flux of Mrk 501 in only two minutes [19].

1.3.2 Super Nova Remnants

Most individual sources of gamma radiation in our own galaxy are the different types of super nova remnants [155, 8], what is left behind by the violent death of a massive star. Depending on the mass of the progenitor star and the age of the SNR, three related main classes exist: pulsars, PWNe and shell-type SNRs.



Figure 1.6: Multi-wavelength image of the Crab Nebula in pseudo-colors. Observations of five different telescopes were combined: VLA (radio, red), Spitzer Space Telescope (infrared, yellow), Hubble Space Telescope (visible, green), XMM-Newton (ultraviolet, blue) and Chandra X-ray Observatory (purple) [119]. The central part, dominated by the purple X-ray observations, shows the pulsar with its accretion disk and jet.

Stars are in an equilibrium between gravitational force trying to collapse them and radiative pressure preventing this. After a star burned through most of its hydrogen fuel, the radiative pressure falls and the star starts to collapse until three helium nuclei can be fused into carbon and pressure is built up again keeping the star stable once more, now burning through its helium

supply. This repeats several times—each time producing heavier elements—until iron is reached, which is the last element where fusion produces energy. If the remaining core is more massive than $1.4M_{\odot}$, the core collapses into a neutron star, if the mass is larger than 3 to 5 times M_{\odot} , it will collapse into a black hole. The collapse also results in the supernova explosion. [57]

Neutron stars are only 10 km to 20 km in diameter and rotate very quickly, because of the conservation of angular momentum and show extreme magnetic fields up to 10^8 T [57]. If the rotation axis is not aligned with the magnetic field axis, neutron stars are observed to emit pulsed beams of radiation. These pulsars, found in most young SNR, are the most numerous galactic source type in the 4FGL catalog [8]. A pulsar can power gamma-ray emission in the surrounding gas cloud left over from the supernova explosion, creating a PWN. The most prominent source of high energy gamma radiation, the Crab Nebula, is a PWN created by a supernova explosion in 1054, observed by Chinese and Japanese astronomers, visible at daylight for 23 days [131]. A pseudo-color image of the Crab Nebula from multiple observatories is shown in Figure 1.6.

Particle acceleration happens very similar to the mechanisms proposed for AGN and the Crab Nebula is particularly well described via a purely leptonic synchrotron and inverse Compton model, using two different electron spectra and four photon distributions: the synchrotron radiation (the self Compton part), thermal emissions from dust, the cosmic microwave background and line emission from the nebula [114]. The full spectral energy distribution from the radio regime to very high energy gamma radiation as measured by several experiments is shown in Figure 1.7. The Crab Nebula also shows a rather constant flux at very high energies and has thus become the “standard candle” of VHE gamma-ray astronomy [114]. The Crab Nebula is used as reference for sensitivities, also in this work, and as a unit of flux: 1 CU being the flux of the Crab Nebula either differential at a certain energy or integrated over a specific energy range. The MAGIC collaboration published a log-parabola parameterization of the Crab Nebula flux valid for the whole energy range observable by FACT in [22]:

$$\Phi(E) = \frac{3.23 \times 10^{-11}}{\text{TeV cm}^2 \text{ s}} \left(\frac{E}{1 \text{ TeV}} \right)^{-2.47 - 0.24 \log_{10}(E/1 \text{ TeV})}. \quad (1.4)$$

This parameterization will be used as reference spectrum to calculate the weights for simulated events in chapter 6.

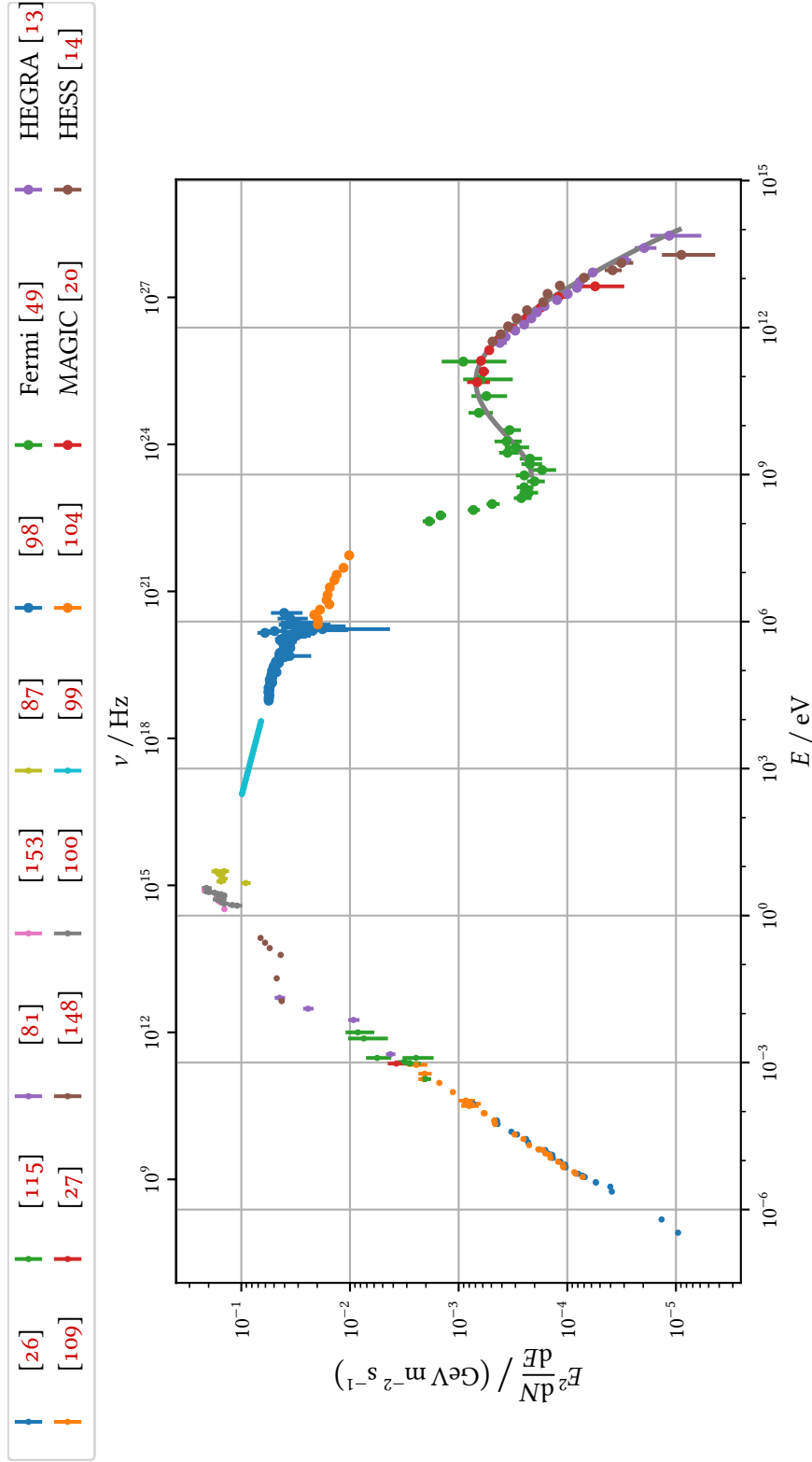


Figure 1.7: Spectral energy distribution of the Crab Nebula, data from 16 observatories, available in a single data file from [78]. The gray line shows the parameterization of the inverse Compton part of the spectrum from [114].

1.3.3 Gamma-Ray Bursts

The third class of gamma-ray sources are the short, transient bursts of gamma radiation which are also of extra-galactic origin. Two classes of bursts exist, bursts with durations longer than two seconds are currently thought to be caused by core-collapse super novae, they make up around 70 % of the detected bursts. One source for the bursts shorter than two seconds was confirmed in 2017, when LIGO and VIRGO observed a gravitational wave event caused by the merger of two neutron stars and a GRB was detected 1.7 s after this signal by the *Fermi* satellite. Later, the kilonova was observed and detected by 70 observatories all around the globe and in Earth's orbit. This first event measured both electromagnetically and in gravitational waves was named "Science Breakthrough of the Year 2017" [4] and was the first true multi-messenger observation.

1.4 Key Science Questions

Stefan Funk [77] and the CTA Consortium [11] identify the following questions as most relevant to currently observing and planned gamma-ray observatories:

The Origin of Charged Cosmic Rays Since first discovered by Victor F. Hess in 1912 [88], the origin of the charged cosmic rays remain largely unknown. Several candidates have been proposed, but none identified with certainty. If gamma-ray sources would be discovered to show spectra only explainable using hadronic models, this would offer a strong hint that those sources also produce charged cosmic rays. This can only be an indirect proof compared to neutrino observations, which would directly point to hadronic reactions.

The Search for Dark Matter High energy gamma-ray astronomy is sensitive to annihilation lines created by several proposed candidates for dark matter particles, e. g. by two weakly interacting massive particles (WIMPs) converting into two gamma rays. Also theories for axion-like particles will be tested via the proposed changes to the mechanics of the extragalactic background light (EBL) absorption.

Understanding Acceleration Mechanisms The precise mechanisms at work in the extreme environments of AGN and pulsars are still a largely open question, especially the very short time scale variability is not described by current models.

Cosmology and Fundamental Physics IACTs are able to test a number of predictions made by several theories of quantum gravity, e. g. Lorentz invariance violations. This is also connected to the question of dark matter.

1.5 Cosmic Rays

Charged primary particles are the main background class for any gamma-ray telescope, also for IACTs. As they are deflected by magnetic fields in the interstellar and intergalactic medium, their origin is not reconstructible at rigidities relevant to IACTs and they thus form an isotropic, diffuse background.

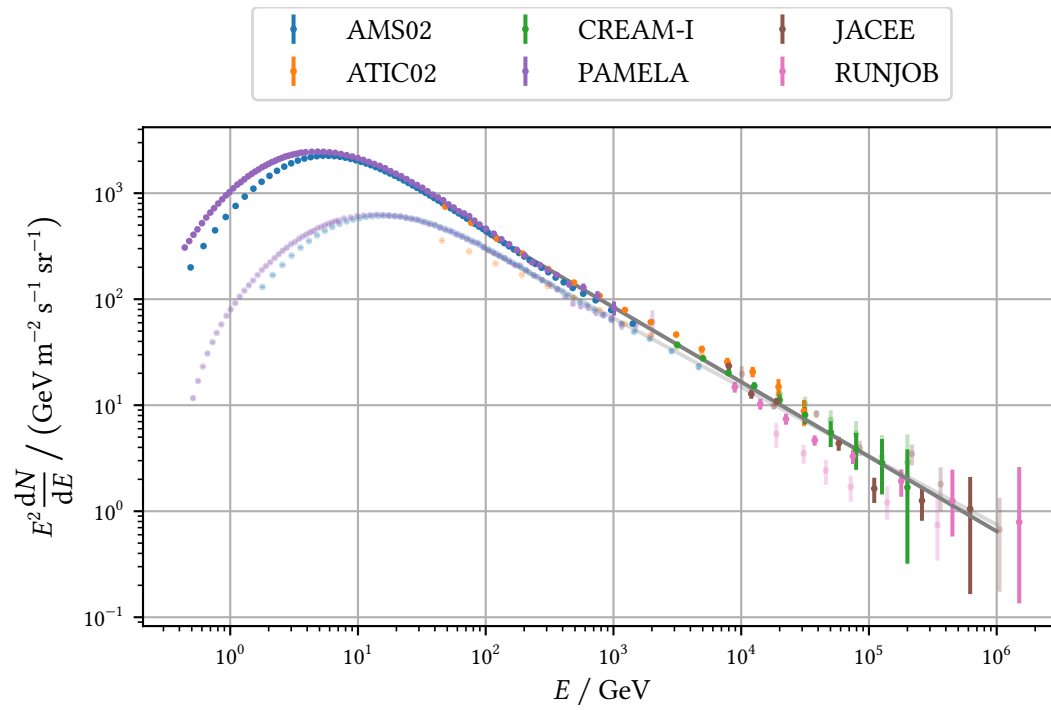


Figure 1.8: Particle flux of protons (solid colors) and helium nuclei (transparent) as measured by several experiments. Data from [111]. The two gray lines are the respective fit results shown in (1.6) and (1.7).

Figure 1.8 shows the flux of primary protons and helium nuclei between 1 GeV and 1 PeV. These two elements make up over 90 % of the charged cosmic rays around 1 GeV [145]. At the energies relevant for FACT, above ~100 GeV, helium nuclei are almost as abundant as protons. The flux of all charged primaries can be approximated above ≈ 100 GeV using

$$I_N(E_N) \approx \frac{1.8 \times 10^4}{\text{GeV m}^2 \text{ s sr}} \left(\frac{E_N}{1 \text{ GeV}} \right)^{-2.7} \quad [145, (29.2)] \quad (1.5)$$

where E_N is the energy *per nucleon*.

The individual spectra for proton and helium nuclei are described using power laws obtained by linear regression of $\log_{10}(I)$ vs. $\log_{10}(E)$ using the data shown in Figure 1.8. For the proton flux, this yields

$$I_p(E) = \frac{10\,922.4}{\text{GeV m}^2 \text{ s sr}} \left(\frac{E}{1 \text{ GeV}} \right)^{-2.705} \quad (1.6)$$

and for the helium flux per nucleus, *not nucleon*,

$$I_{\text{He}}(E) = \frac{5922.9}{\text{GeV m}^2 \text{ s sr}} \left(\frac{E}{1 \text{ GeV}} \right)^{-2.650} \quad (1.7)$$

is obtained. The covariance matrices for the fits are given in Appendix 3.

For an observation of the Crab Nebula, the ratio of gamma rays from the source to cosmic ray background events will approximately be

$$\frac{N_Y}{N_{\text{CR}}} = \frac{\int_{E_{\text{min}}}^{E_{\text{max}}} \int_A \int_{t_0}^{t_1} \Phi(E) dt dA dE}{\int_{E_{\text{min}}}^{E_{\text{max}}} \int_A \int_{\Omega} \int_{t_0}^{t_1} I_N(E) dt d\Omega dA dE} \quad (1.8)$$

where Φ is the flux of the Crab Nebula, which for most IACTs can be assumed to be a point source and thus no integration over the solid angle is necessary. We can now calculate this ratio for an IACT with a field of view with a diameter of 4.5° , resulting in $\Omega = 2\pi(1 - \cos 2.25^\circ) \text{ sr} \approx 0.005 \text{ sr}$. The area of the detector and the observation time is the same for both terms and cancels out.

$$\frac{N_Y}{N_{\text{CR}}} = \frac{\int_{E_{\text{min}}}^{E_{\text{max}}} \Phi(E) dE}{0.005 \text{ sr} \int_{E_{\text{min}}}^{E_{\text{max}}} I_N(E) dE} \quad (1.9)$$

using the log-parabola parameterization of the Crab Nebula flux published by the MAGIC collaboration in [22] and an energy range of 100 GeV to 200 TeV results in

$$\frac{N_Y}{N_{\text{CR}}} \approx \frac{1}{4300} \quad (1.10)$$

So for each gamma-ray induced shower, 4300 showers induced by charged cosmic rays are reaching the telescope. We will see in section 6.5 that FACT can detect sources with down to 10 % of the flux of the Crab Nebula in 50 h of observation time, MAGIC is an order of magnitude more sensitive reaching around one percent in 50 h [21]. This directly translates into the ratio of signal to background events and will necessitate an analysis that can classify recorded events as either gamma-ray or cosmic-ray induced to suppress this background as much as possible. Methods to achieve this will be introduced in chapter 5.

The First G-APD Cherenkov Telescope



Figure 2.1: FACT in front of the MAGIC-II telescope at the Observatorio del Roque de los Muchachos. The white cylinder on the left is the camera, complete with all readout electronics.

In this chapter, I will introduce the necessary aspects of the FACT telescope, which are needed to understand the simulation and analysis presented in the next chapters.

After a general introduction, I will focus mainly on the data acquisition and electronics, as these are the parts of the telescope that are most crucial for my work on improving preprocessing of raw data and simulations.

One of the main goals of FACT was to demonstrate the possibility of using Silicon Photo Multipliers (SIPMs) for Cherenkov astronomy [25]. It has been the first telescope to employ this new technology of photodetectors in the field when it started operating in 2011. Now, eight years later, all prototypes for the Small Sized Telescope (SST) for CTA are based on SIPM cameras and the final design was agreed upon in 2019 [159]. It can be safely concluded that FACT reached this goal.

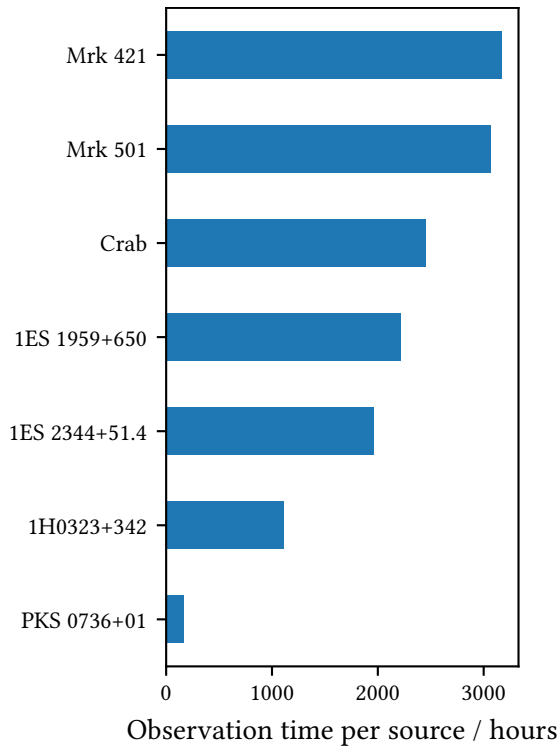


Figure 2.2: Observation time for the seven sources most observed by FACT. Together they amount to over 95 % of the total time the telescope has observed.

Ray Astronomy (HEGRA) CT-3 telescope, equipped with new drive motors, mirrors and the SIPM camera. Compared to the other currently operating telescopes, like VERITAS or MAGIC, FACT is rather small, limiting its sensitivity to the brightest sources. It is also the only currently operating monoscopic Cherenkov observatory. FACT’s reflector is a segmented mirror with 30 individual hexagonal facets, creating a total reflective surface of 9.5 m^2 with approximately four meters in diameter. Each mirror has a focal length of 4.9 m, which is also the effective focal length of the segmented reflector. The facets were oriented in the Davies–Cotton–Design[55] before this was changed to a hybrid between Davies–Cotton and a parabolic orientation in May 2014, as a compromise between spatial and temporal resolution of the reflector system [16]. At this point, the mirrors were also realigned, resulting in an overall better resolution than before.

2.1 Photosensors and Readout Electronics

FACT’s camera comprises 1440 pixels, each consisting of a 3 mm by 3 mm SIPM that is subdivided into 3600 individual Geiger-Mode Avalanche Photo Diode (G-APD) cells. G-APDs are photodiodes operated with a reverse bias above the breakdown voltage, the limit above which a single electron/hole pair created by an absorbed photon can induce an avalanche. The cell is connected in series to a resistor and the current created by the avalanche results in dropping the bias below the

Compared to the traditionally used PMTs, these semiconductor photosensors are much more robust to high light levels. Due to this, FACT can observe under light conditions, e. g. in moonlit nights, where other IACTs can not [101]. This makes FACT uniquely suited for its main physics goal, to monitor the brightest sources of gamma rays as densely as possible, which is also evident in Figure 2.2, showing that FACT spends most of its observation time on just seven sources. Shortly after starting observations in October 2011, the FACT collaboration started publishing results of a “quick look analysis” [62] as soon as they were available. This analysis provides gamma-ray event rates normally around 20 minutes after the last event of a run was taken. Based on this analysis, the FACT collaboration has issued several alerts to the other IACTs to inform them of unusually bright flux of the monitored sources [63].

FACT is built upon the refurbished mount of the former High Energy Gamma-

breakdown breakdown, stopping the avalanche. After an avalanche, the cell needs a short time to reset to the full bias voltage again. During the dead time, while the cell is below the breakdown voltage, no new signal can be created. While it is above the breakdown but below the nominal operation voltage—called recovery time—, the signal is lower than normally [25]. The single cells are connected in parallel to form a single, accumulated output signal per pixel. SIPMs do not need a high voltage supply in the kV regime like PMTs do. FACT’s photosensors have a breakdown voltage around 70 V and are operated at an overvoltage of 1 V, while some more recent models even only require 30 V [24].

However, SIPMs do not come without issues. One disadvantage is the temperature dependent gain of SIPMs and two main approaches exist to deal with this: active cooling of the sensors to a fixed temperature or a system adjusting the bias voltage of the SIPMs so the gain stays stable. FACT has implemented the second path, only relying on removing excess heat from the camera, not keeping a fixed temperature for observations. The bias voltage is generated in the container next to the telescope and supplied to groups of 4 or 5 pixels, called bias patches shown in Figure 2.4. [25]

Compared to PMTs they have a higher dark count rate, signals induced not by photons but by thermal electrons in the sensor itself. Cross talk, the effect that a break through in one cell can trigger another in a neighboring cell, results in a signal corresponding to two photons or more. There is also a chance that a discharge triggers a later discharge of the same cell, which is called an after pulse. [65]

To increase the light gathering surface area, a Winston cone with a hexagonal entry window with an incircle diameter of 9.5 mm is placed in front of each pixel. The resulting camera is roughly 40 cm in diameter which, together with the 4.9 m focal length of the reflector, gives a field of view of 4.5° . Each individual pixel covers 0.11° in diameter on the sky.

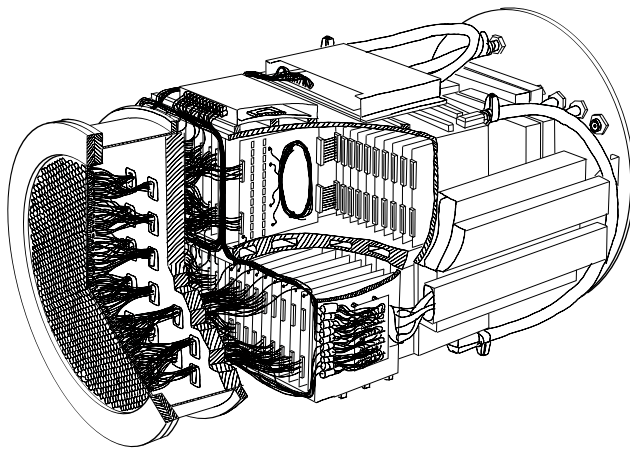


Figure 2.3: Schematic view of the camera. From left to right: the 1440 pixels in the sensor compartment, the insulation between sensor compartment and readout electronics, the 4 crates with the preamplifier boards and the fast analog-to-digital converter (FADC) boards. The FACT trigger master (FTM) can be seen at the top next to the insulation. [Graphic by Sebastian A. Mueller]

The analog signal of the pixels is wired to the preamplifier boards, that handle 36 pixels each and are organized in four crates of ten boards of the preamplifier and FADC boards each. The trigger unit on the preamplifier board clips and sums the signal of 9 pixels each, these groups are referred to as trigger patches. The preamplifier board applies an N -out-of-four logic to these trigger input signals, which is the input for the FTM board, which applies a N -out-of-40 logic to decide if an event should be stored to disk. For normal physics observations, both N are 1, thus if any of the 160 trigger patches is above the trigger threshold, an event is stored to disk. In addition

to events triggered by this system, additional events are recorded during normal data runs. The camera is read out once per second regardless of a trigger signal, these are so called interleaved pedestal events which can be used to estimate the noise level. Since November 2013, the camera is also triggered at every full second by a GPS receiver to provide more precise timing information. Until 2017, an LED light pulser in the reflector dish fired every few seconds, this was done for calibration of the pixel response. However, this LED light pulser went rogue in 2015 and also fired without being activated creating large events triggered by the physics trigger until it was finally removed in April 2016.

The FADC boards also receive their signal from the preamplifier boards. The heart of the FADCs are the Domino Ring Sampler 4 (DRS4) chips [137]. These chips sample the analog signal using 1024 capacitors for each channel using a sampling frequency of 2 GSample/s. In case the FTM issues a trigger decision, the voltage stored in the capacitors is readout and digitized by a 12-bit analog-to-digital converter (ADC). For regular observations, only 300 of the samples are readout, starting 50 samples before the trigger signal was received [25]. While the DRS4 chips allow fast digitization of the SIPM signal, it also introduces the need for several calibration steps and adds artifacts to the data [142]. The most common artifacts are sudden increases of the digitized signal in a single or two adjacent cells called spikes. These interfere with the searching for the Cherenkov pulse signal if not corrected. Different amounts of remaining charge in the cells are responsible for sudden shifts of the baseline for longer time periods when reading out cells that have had different times since their previous readout. In FACT's terminology, these are called jumps.

The digitized samples are transferred via four ethernet connections to the data acquisition computer which writes the data to disk. Under good environmental conditions FACT triggers between 60 and 80 events per seconds [35, 25]. The organization of the FACT camera pixels into the different groups is shown in Figure 2.4.

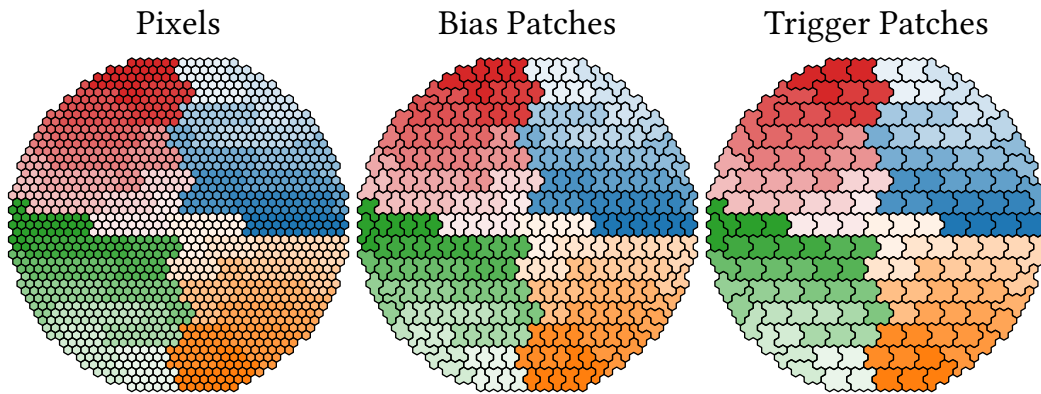


Figure 2.4: Organization of the FACT camera. The four main colors highlight the crates, the variation in those colors the boards. The left image shows the individual pixels. In the central image, pixels using the same bias voltage supply are grouped. The right image shows all pixels on the same DRS4 chip, which also corresponds to the trigger patches.

2.2 Observation Strategy

As not all recorded background events can be discarded reliably while still keeping most of the gamma rays, a method to estimate this residual background is needed. A simple method would be to observe the gamma-ray source first and then observe a similar region in the sky without a gamma-ray source for the background estimation. This has several important drawbacks, first the available time for source observations is cut in half. Second, the observation conditions might change and thus bias the estimation of the background.

A way around this is not to position the source of interest in the center of the camera but using a small offset. In this case, several regions in the camera geometrically equivalent to the position of the gamma-ray source can be used for the background estimation. This maximizes observation time for gamma-ray sources, guarantees identical observation conditions for the background estimation and also decreases the statistical uncertainty for the background, as several positions can be used [76]. The situation is sketched in Figure 2.5. Nearly all FACT observations are carried out in this manner. Due to the characteristic switching between positions relative to the source, making small jumps around its position, this strategy is called “wobble mode”.

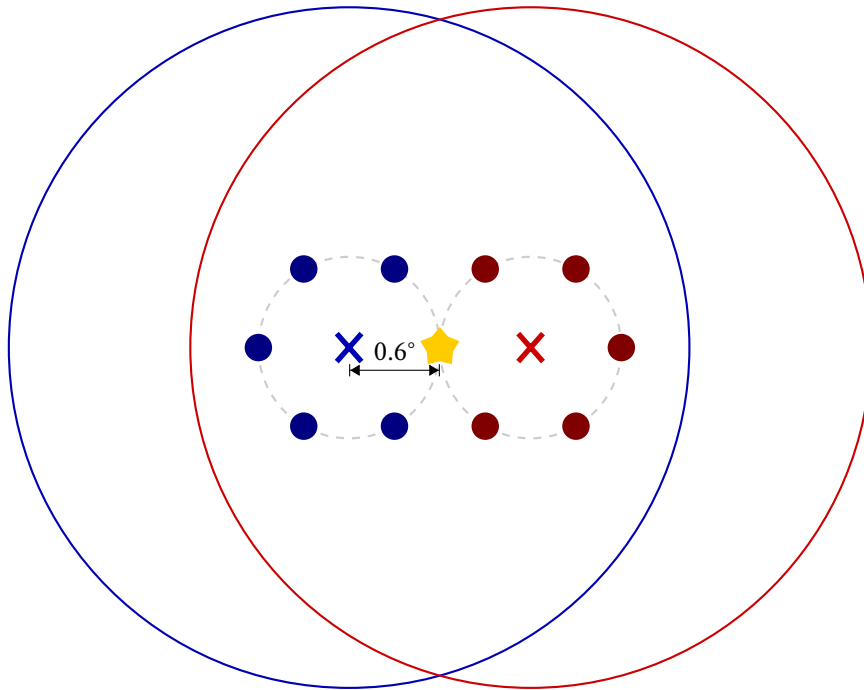


Figure 2.5: When observing a source in wobble mode, the source (yellow star) is not in the center (crosses) of the field of view but offset by a certain amount. FACT uses 0.6° . Shown here are two wobble positions at wobble angles of 0° (red) and 180° (blue) as measured from the right ascension axis. FACT normally observes using two wobble positions at 180° offsets, changing position every twenty minutes. The bold circles show the five off positions per pointing position that are used to estimate the background. Because there is a bright star in the field of view for Crab Nebula observations, the wobble angles are shifted by 50° , so the star is not close to any of the off positions.

Monte Carlo Simulation of FACT Events

To evaluate the performance of an instrument design before it is build or to generate labeled data for the training of statistical models, the possibility of simulating the data of instruments is of crucial importance. As there are no artificial sources of gamma rays and cosmic rays at the energies measured by IACTs, extensive simulations are necessary to be able to reconstruct the events observed with these telescopes.

Simulation of events for IACTs is usually done in two steps: First, the EAS is simulated which produces the Cherenkov light on the ground level at the telescope as output. Second, from the simulated Cherenkov photons, the detector response of the instrument is calculated.

Due to the stochastic nature of the involved processes, the Monte Carlo method is the perfect fit to generate artificial IACT events with known initial conditions. Very few initial parameters result in an incalculable large number of possible outcomes. Simulating a single air shower requires drawing several million pseudo random numbers.

For small instruments like FACT, the simulation of the EAS is usually much more computationally intensive than the simulation of the detector response.

3.1 Simulation of Air Showers Using CORSIKA

The development of Cosmic Ray Simulations for Kascade (CORSIKA) [84] initially began in the late 1980s by merging three precursor code bases into a single software to simulate EAS for the KASCADE [60] cosmic ray experiment. It was later extended to also be able to calculate Cherenkov light production, currently with the `iact/atmo` extension [33].

CORSIKA simulates all particle interactions from the moment a particle enters Earth's atmosphere. This requires knowledge of the cross-sections for all possible processes, especially for very high energy interactions in forward direction and for air molecules. These regimes are generally not accessible by collider experiments and are thus extrapolated from the available data at lower energies, larger scattering angles and other nuclei [133]. Improvements to the cross-section calculations after new experimental data is published is one of the main improvements between CORSIKA versions.

To produce simulations with CORSIKA, a large number of configuration options have to be considered, both at compile and run time. These need to be adapted for the specific experiments and several libraries can be chosen for the calculation of the particle interactions.

CORSIKA is written in FORTRAN 77, while the `iact/atmo` extension is written in C. Unfortunately, CORSIKA is not Free and Open Source Software. However, access is usually granted for anyone interested in using the software. There is an ongoing effort to create the next major version of CORSIKA using more modern technologies in a complete redesign, openly developed by the

community and published under an open source license. However, this is not yet finished as of early 2020 [66].

In this work, new simulations were produced using CORSIKA, version 7.69 released in December 2018 of CORSIKA. These are compared to simulations that were previously created by the FACT-Collaboration using a customized version of CORSIKA called MAGIC Monte Carlo Software (MMCS), which was based on version 6.5 [144], released in 2006.

3.1.1 Interaction Models

The particle interaction cross sections are not calculated by CORSIKA itself but by externally contributed libraries. When simulating Cherenkov light, the electromagnetic interactions are simulated using Electron Gamma Shower 4 (egs4) [34].

For the hadronic interactions, several libraries are available for use with CORSIKA. As these libraries have their own energy domains, CORSIKA splits calculations into two energy regimes each handled by a different library with a hard transition at 80 GeV [85]. Choosing the libraries for the different cross section calculations is a compile time option.

There are three possible choices for the hadronic interactions below an energy of 80 GeV in CORSIKA for both versions 7.69 and 6.5: Gamma Hadron Electron Interaction Shower (GHEISHA), Fluctuating Cascade (FLUKA) and Ultra-relativistic Quantum Molecular Dynamics (URQMD) [85]. According to a comparison of the three models in [64], GHEISHA fails to reproduce measured data when used with CORSIKA to simulate air showers. It was thus not considered for the simulation of FACT events. FLUKA [69] is only provided as compiled binary to registered users with a very restrictive license [74]. These binaries even have a “shelf life” date compiled into them, after which the software will stop to work. As this is all but prohibiting the reproducibility of the simulations, FLUKA should not be used for a new set of simulations. However, for a long time it was considered to be the best model [64] and it was used to produce the simulations with MMCS 6.5. URQMD [28], the third option, will be explored as an alternative to FLUKA.

The possible choices for the hadronic interaction models above 80 GeV are more numerous, with some of the options being combinations of the other models. The MMCS 6.5 production used Quark Gluon String model with Jets (QGSJET). With the advent of Large Hadron Collider (LHC) measurements, many libraries were updated to reproduce the now available data at higher energies and for lead collisions [133]. Together with the `iact/atmo` extension discussed in the next section, these are the major changes from using MMCS 6.5 to CORSIKA 7.69. The most advanced model is currently EPOS-LHC [133], which combines the older VENUS and QGSJET models with better extrapolation and LHC data.

3.1.2 Cherenkov Light Production

Any charged particle traveling faster than the local speed of light induces emission of Cherenkov light. CORSIKA divides the tracks of particles into steps between interactions, decays or the deflection in Earth’s magnetic field. These track parts along with start and end energy of the particle are handed over to the subroutines calculating the Cherenkov emission.

A first version of the Cherenkov production code, used by the MMCS 6.5 production, was developed for non-imaging detectors of the HEGRA array and proved to be not ideal for IACTs. The later developed `iact/atmo` extension improves several aspects: photons are collected in

spheres encompassing the telescopes instead of rectangles on the ground, better description of the atmosphere by using fine-grained, tabulated values for density and refractive index and taking atmospheric refraction of the Cherenkov photons into account [33]. Due to these improvements, the `iact/atmo` extension will be used for the new simulations using CORSIKA 7.69. The `iact/atmo` extension writes its output in the `eventio` format [32], a generic container format for binary payloads, which are defined by the applications writing and reading the data. Two different formats exist for the Cherenkov payload, a smaller representation called “compact” with limited precision using 16-bit integers to store scaled quantities and a larger storing 32-bit floating point values.

Both the `iact/atmo` extension and `MMCS` offer the possibility to *reuse* a shower. Meaning that the same telescope is randomly placed at different positions in the Cherenkov light pool of a shower, effectively creating multiple events from the same air shower. This results in a very direct speed up of simulations and since most of the simulated showers will not trigger the telescope or survive the event selection in the analysis, it is quite rare that the same shower is used multiple times in the end, which could introduce a statistical bias as events would not be independent anymore. Because computation time and small event numbers are mainly a problem of hadronic simulations, a reuse of 20 is only used for these showers and not for photon primaries. Another option of improving simulation performance is treating several photons as a single ray with a weight corresponding to the number of photons. This is called bunching and is a compromise between accuracy of the spatial distribution and computing time.

Because FACT’s photo detectors are sensitive to larger wavelengths than classical PMTs, Cherenkov photons are simulated in wavelengths from 290 nm to 900 nm.

3.1.3 Particle Properties and Other Configuration Options

All runtime options for an individual corsika run, meaning a single execution of the program simulating a fixed number of showers, are configured in a plain text file called the input card. This includes properties of the primary particles, description of the telescopes, seeds for the random generator to ensure reproducibility, the local magnetic field, the atmosphere and many more.

The initial parameters of any particle in the simulation are

- Particle type
- Energy
- Direction
- Position, expressed as the impact point on the ground for primaries.

At the beginning of each new simulation event, these properties are either fixed or drawn randomly from the corresponding distributions. In the simulations done for FACT, the particle type is always kept fixed, either gamma rays, protons or helium nuclei are simulated. The energies of the simulated events are drawn from a power law distribution with spectral index γ between a minimum (E_{\min}) and a maximum energy (E_{\max}). The distance to the telescope R , also known as the impact parameter, is drawn so that the events are inside a circle of radius R_{\max} and uniform in area. The direction is sampled uniformly in the azimuth angle φ between φ_{\min} and φ_{\max} and the zenith angle ϑ is sampled to be uniform in solid angle between ϑ_{\min} and ϑ_{\max} . The corresponding probability density functions are given in Appendix 4.

3.2 Simulation of Detector Response using CERES

The detector simulation of FACT is done using Camera Electronics and Reflector Simulation (*ceres*) [45], an executable using FACT's version of the Magic Analysis and Reconstruction Software (MARS)¹ framework to simulate the detector response given Cherenkov photons on the ground as input. *ceres* supports both the classical CORSIKA binary output format and the *eventio* output format produced by the *iact/atmo* extension as discussed in the [previous section](#). Reading of *eventio* however is rudimentary and does not support all features of CORSIKA, e. g. only a bunchsize of 1 and the compact format of the CORSIKA output are supported. As with CORSIKA, *ceres* has a multitude of configuration options, describing the telescope, the pointing position relative to the shower axis and the environmental conditions.

The first step in the simulation in *ceres* is the application of several absorption processes, reducing the amount of photons the simulation needs to process. As each photon that finally might be detected in one of the telescope's pixels goes through the same processes that might absorb the photon, all of these processes can be combined into an overall detection efficiency, greatly reducing computational cost through applying this absorption as early as possible. These absorptions include atmospheric scattering, reflectivity of the mirrors, transmittance of the entrance window of the camera and the Winston cones and the photon detection efficiency of the SIPMs. All these efficiencies are tabulated vs. wavelength in files as input to the simulation. For the simulations produced for [149], an additional absorption without direct physical cause further reducing the number of photons was introduced. This is used to model all neglected sources of photon loss, like shadowing effects of the mast structure holding the camera.

After discarding photons according to the overall absorption, simplified ray tracing is done to simulate the optical system, calculating which pixel if any is hit by the photon. The segmented mirror is defined in a configuration file, providing shape, size, focal length, position and orientation of each facet.

The signal of the SIPMs is calculated by superimposing the pulse shape of a single photon for each individual photon, also adding additional photons for the Night Sky Background (NSB), dark counts and cross talk. The pulse shape can either be defined using tabulated values in a text file or an analytical function. An additional electronic noise, only simulated as simple white noise is added to the resulting signal.

Several options to further add artifacts to the simulated timeseries were implemented for the simulations produced for [149]. It was observed that the temporal features are not well-described, especially the variance of photon arrival times was much smaller for simulated events than observed on measured data. Three different approaches were introduced to tackle this problem: a fixed time offset for each pixel, so that the signals would go out of sync, a random offset between pixels for each event and smearing the arrival times of each individual photon by a normally distributed random number.

The signal is then used to simulate the trigger decision and if any trigger is raised, the event is stored to disk. *ceres* supports both ROOT output as well as the Flexible Image Transport System (FITS) format, which produces files very similar to the ones the actual telescope data acquisition writes.

¹While sharing a common history, development of FACT's and MAGIC's version of the software split before FACT started operation in 2009. MAGIC's version is not public.

The MARS/ceres version used for the simulations done in this work is revision 19561 in FACT's subversion repository [67].

3.3 Reweighting Simulated Events to Physical Fluxes

To compare observed event distributions to simulated ones, to calculate sensitivities for a given gamma-ray source or to fit flux models to observed events, a simulated event distribution needs to be reweighted to represent a physical flux.

The first step is to calculate the flux normalization of the simulated events Φ_{sim} for a given observation time t_{obs} . The number of simulated events N is the integral of the differential flux:

$$N = \int_{E_{\text{min}}}^{E_{\text{max}}} \int_0^{t_{\text{obs}}} \int_{\Omega} \int_A \Phi_{\text{sim}} \left(\frac{E}{E_{\text{ref}}} \right)^{\gamma} = \frac{\Phi_{\text{sim}} t_{\text{obs}} \Omega A}{E_{\text{ref}}^{\gamma} (\gamma + 1)} (E_{\text{max}}^{\gamma+1} - E_{\text{min}}^{\gamma+1}). \quad (3.1)$$

Solving for Φ_{sim} yields

$$\Leftrightarrow \Phi_{\text{sim}} = \frac{E_{\text{ref}}^{\gamma} (\gamma + 1) N}{t_{\text{obs}} \Omega A (E_{\text{max}}^{\gamma+1} - E_{\text{min}}^{\gamma+1})} \quad (3.2)$$

Where $A = 2\pi R_{\text{max}}^2$ and Ω is the solid angle, which is given by

$$\Omega = 2\pi(1 - \cos \theta_{\text{max}}) \text{ sr} \quad (3.3)$$

if the events were simulated using a viewcone and $\Omega = 1$ if a point-like source was simulated.

The sample weights for each simulated event are now calculated as the ratio of the target energy spectrum and simulated energy spectrum

$$w_i = \frac{\Phi_{\text{target}}(E_i)}{\Phi_{\text{sim}}(E_i)} \quad (3.4)$$

3.4 Monte Carlo Production Software

To run the large number of tasks required for the Monte Carlo simulations, a software library and a set of executable programs were developed, with the following requirements identified:

- Templating for the configuration files of the simulation programs, so that job parameters can be stored centrally.
- Tracking of job status, input and output files.
- Automatic installation and compilation of the needed software.
- With the above, also the possibility to use several compiled versions of CORSIKA and ceres to enable comparisons.
- Possibility to run computation tasks on multiple backends, at least on the SLURM [143] cluster engine and locally using multiple CPU cores.
- Reproducibility of jobs.
- Central job queue optionally with multiple consumers.
- Collection and compression of simulation results.

3 Monte Carlo Simulation of FACT Events

- Possibility to use multiple database engines, at least `sqlite` for easy, local processing and testing and a fully grown relational database like MySQL or PostgreSQL for larger productions.

The software called `mopro3` [123] was implemented in python. Two similar projects existed before, but were centered around the older `MMCS` and had several shortcomings with respect to the requirements listed above, lacking support for automatic installation of software, multiple consumers, running jobs locally and on multiple sites.

The job configuration and simulation settings are stored in a relational database, using the `peewee`[106] Object Relational Mapper. The relational mapper allows access to the database from python as objects and supports all major database systems.

Templating of the configuration files is done using the `jinja2` templating library, an example of a `CORSIKA` input card template can be found in Appendix 6, the `ceres` template in Appendix 7. The expressions in double-curly braces are evaluated and replaced for each database entry before giving it as input to `CORSIKA` and `ceres`, respectively.

Two computation backends are currently implemented. First, jobs can be submitted on a cluster using `SLURM`, which is used for job management at the `LIDO3` cluster in Dortmund and the `LESTA` cluster in Geneva, where `FACT`'s data is stored. Jobs can also be processed using local worker processes, which eases development and running small simulation productions. Other backends can be added by implementing a small number of required methods, e. g. for submitting, canceling and getting the status of jobs.

The main program queries the database in certain intervals for pending jobs, and if possible submits them for computation to the workers. An extra thread in the main program uses a `ØMQ` [89] server socket to receive job status updates which are then persisted to the database.

The simulation programs `CORSIKA` and `ceres` are wrapped in small scripts to copy input files to the local working node, handle failures and communicate status transitions to the main program as mentioned above.

When a new job is submitted to the workers, the main program makes sure all necessary software is installed, creates the required directories for output and log files and creates the configuration files from the templates and parameters stored in the database.

As any `ceres` job depends on the successful `CORSIKA` job it uses as input, `ceres` jobs are started when their corresponding `CORSIKA` job finishes successfully and it was computed at the same location.

`mopro3` itself is configured using a `yaml` file. The settings include credentials for downloading `CORSIKA`, the database configuration, which temporary directory to use for job processing and options for the workers like which queues to use on a `SLURM` cluster or how many CPUs to use in parallel for local processing.

3.4.1 Database Structure

The `mopro3` database contains five tables, one describing general settings for `CORSIKA`, one for each individual `CORSIKA` run, the same two tables for the `ceres` jobs and one table describing the job states.

For the `CORSIKA` settings and runs, a compromise between flexibility and complexity of the database and implementation was made. `CORSIKA` has both compile time and execution time options.

The compile time options are normally set interactively by executing the `CORSIKA` configuration utility (`coconut`) program. This creates a header file `config.h` defining preprocessor variables that result in conditionally compiling the corresponding code. This is used among other things for the interaction models, thus the same `CORSIKA` installation cannot be used with different interaction models. Instead of implementing all possible compile time options in the `CorsikaSettings` table and generating the `config.h` file at installation time, the user has to generate the file using `coconut` and the full file is stored in the database and used to compile `CORSIKA`. This also has the advantage of being rather agnostic about the version of `CORSIKA`.

A similar approach is taken for the large number of runtime configuration options `CORSIKA` supports, which in turn also depend on the compile time configuration. Only the most fundamental settings most likely to change between runs of the same kind are stored in the database. Other parameters have to be changed via the input card template in the `CorsikaSettings` table. This also has the advantage that they are very unlikely to change with a new release of `CORSIKA`, as all these options are very fundamental, improving maintainability of `mopro3`. In most cases, no changes should be required when a new minor version update of `CORSIKA` is released.

The `Status` table stores all possible states a run can reside in. All states and their possible state changes are shown in [Figure 3.1](#).

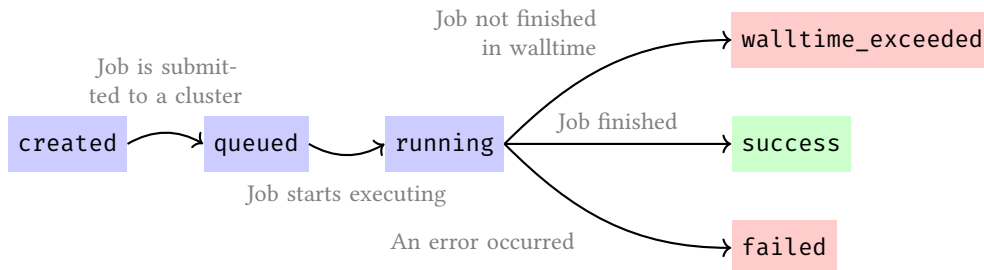


Figure 3.1: Possible state transitions for the `mopro3` processing. Blue indicates unfinished jobs, green successfully finished jobs and red failure modes.

3.5 Comparison of FLUKA and URQMD

As described in section 3.1.1, there are two options to consider for the low energy hadronic interaction model. To compare the two, a smaller test data set using protons as primary particle was produced for each `FLUKA` and `URQMD`.

The full simulation and analysis chain up to the image parameters as described in [chapter 4](#) was executed. [Figure 3.2](#) shows the distributions of several image parameters for the two new simulation sets and the existing set produced with `MMCS 6.5`. While the new version of `CORSIKA` shows some differences compared to the older version, the differences between the two interaction models are negligible on the image parameter level. Considering the problematic license of `FLUKA` and that `URQMD` also runs simulations about twice as fast, it was decided to use `URQMD` for the new production of `FACT` simulations.

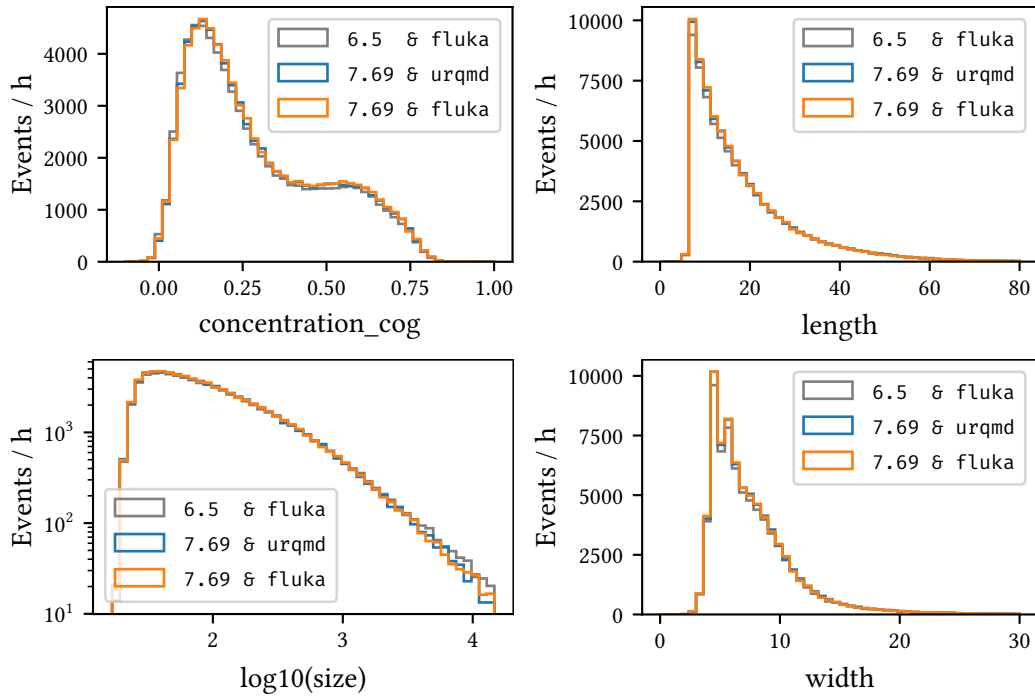


Figure 3.2: Image parameters concentration_cog, length, size and width for the three different simulation data sets. The differences between URQMD and FLUKA are negligible while there is a slight difference between the two and the older simulation settings, e. g. for large values of size and small values of concentration_cog.

From Raw Data to Image Parameters

The topic of this chapter is the path from FACT's raw data, be it measured by the telescope or simulated by software, to a small number of image parameters.

This is a large reduction in the amount of data, from several hundred terabytes of raw data for all recorded events since FACT started observing, to under two hundred gigabytes for the image parameters. These image parameters will later be used to estimate the physical properties of the primary particle that induced the air showers as described in [chapter 5](#).

4.1 Data Format and Structure

FACT's data acquisition system writes one file per so called *run*, which under normal data taking conditions corresponds to five minutes of observation time. During twilight, FACT will also use one minute long runs. Several other types of runs, mostly for calibration purposes, do not have a fixed observation time but a fixed number of recorded events.

The raw data of a single event consists of the digitized voltage series for each of the 1440 pixels. For physics runs, 300 values for each pixel are stored, corresponding to a total time of 150 ns. Due to the trigger configuration, the main signal of a Cherenkov shower is expected around 25 ns and is usually around 50 ns long. In the end of the time series, a calibration signal is fed into every ninth pixel, which is however not used in the analysis.

In addition to the voltage series, metadata is provided, including the time the event has been observed and the physical capacitor first digitized for each DRS4 chip, which is needed for the calibration. A typical FACT data run contains around 20 000 events.

The raw data is stored in a custom extension of the FITS[157] file format. This custom extension provides faster and higher compression [15] than for example `gzip`, but results in files that are not fully compliant with the current FITS standard [73]. Thus custom software is required to read the telescope's raw data from the *binary table extension* that was compressed using the algorithms described in [15]. To make things worse, the description in [15] differs in subtleties from the actual implementation in the FACT data acquisition software available in [44]. Simulated raw data is stored in standard-compliant FITS files using the same general structure, but containing additional information about the simulations, like the energy or type of the particle that induced the air shower.

The analysis also needs additional information not available from the raw data input files. This so called auxiliary data is stored in a FITS file each per telescope subsystem and per observation night. The frequency in which information is stored into these files is different for each subsystem and depending on the type of information, either the closest data point, the last data point or an interpolation has to be used to derive the needed information for the event at hand. Auxiliary data include the current pointing position of the telescope, which is recorded by the telescope's

drive every two seconds, and the currently observed source, which is only reported when a new observation is started.

4.2 FACT-Tools

All steps described in this chapter will be carried out using the software FACT-Tools[48]. This software is written in Java and was developed in Dortmund starting in late 2012 in close cooperation with the computer science department as part of the Collaborative Research Center 876. The FACT-Tools are an extension of the streams framework [38] to analyze the raw data of FACT. A large part of this work was concerned with improving this software in various aspects. Between version v0.15.0, which was released in May 2016 and version v1.1.3, which was released in October 2019, a total of 1232 commits¹ were made, of which 552 were contributed by the author of this thesis.

The streams framework allows for modelling of data flow graphs using xml definitions. A Stream creates new data items, e. g. by reading from an input file. Each data item is then handed over to a Process that executes a number of Processors that get the data item as input and return it as output, doing calculations and adding new members to it in between. A Processor can also track state between data items and execute code after the Stream has finished. Services provide out-of-flow data to Processors, for example calibration constants for the events that are not stored in the input files themselves. The FACT-Tools define and implement Streams and Processors that read FACT data and apply the necessary analysis steps to process the data all the way from raw time series for each pixel to image parameters per event. The xml file for steering the FACT-Tools standard analysis used for analysing observed data is available in Appendix 9. It includes other xml files, that each perform the steps detailed in the next sections. The FACT-Tools are compiled and bundled together with all dependencies into a single executable jar file [130], which is platform independent and only requires the Java virtual machine in version 8 installed. The jar files are produced and uploaded automatically for each released version.

4.3 Raw Data Calibration

As described in section 2.1, the FACT camera digitizes the analog signal of the pixels using DRS4 chips, which introduce several artifacts that have to be taken care of. First, the amplitude of each voltage sample is corrected for their temperature dependence and production differences using calibration constants from the DRS4 calibration files that are generated several times during the night [35]. This step also converts the time series back from ADC counts to voltages. After the amplitude calibration, jumps and spikes are removed. In the final step of the drs4 calibration, the time series are corrected for the not perfectly uniform sampling. Jumps and non-uniform sampling are not simulated by CERES, thus these calibration steps are only applied to observed data. Figure 4.1 shows the time series of a pixel before and after the calibration.

¹A commit is a single set of changes made to a collection of files, tracked by a Version Control System.

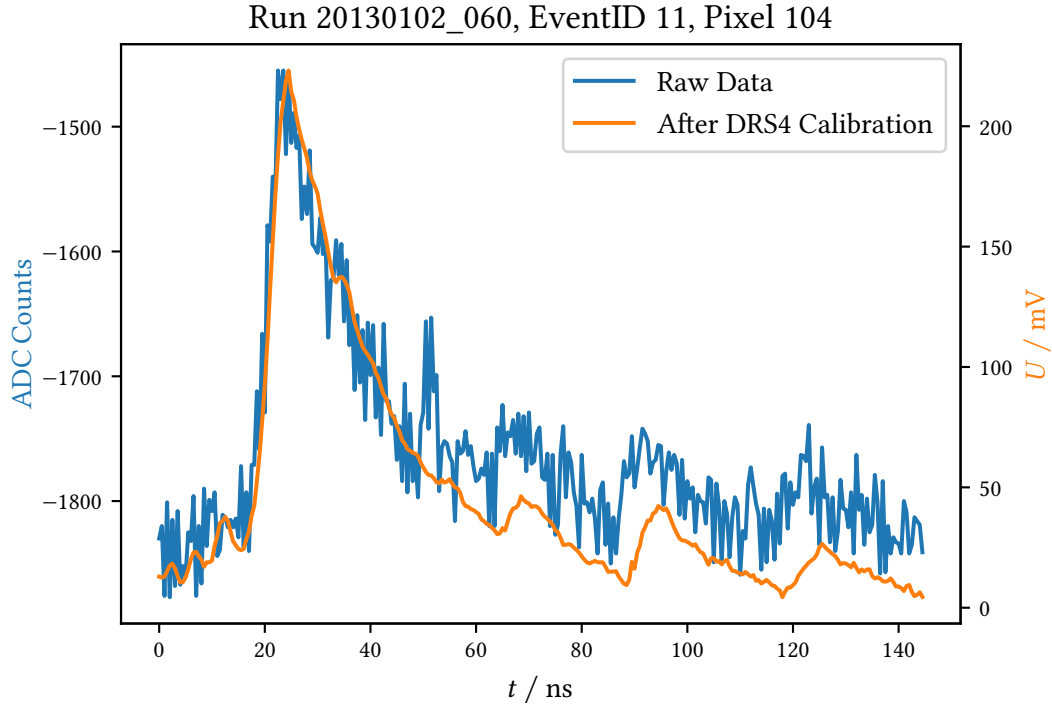


Figure 4.1: Time series of a single FACT pixel before (blue, left y-axis) and after DRS4 calibration (orange, right y-axis).

4.4 Signal Extraction

In a first data reduction step, the time series per pixel are reduced to an estimated number of photons and their mean arrival time for each pixel. This is done by integrating the pulses over 30 time slices starting at the half height of the maximum. The conversion factor from this integral to the number of photons, called gain, is taken from “single p. e.” measurements, runs with closed camera lid, so no light can reach the photosensors, where FACT reads out a fixed number of events, usually 10 000. These runs are taken at the beginning and end of each night. Due to dark counts and cross talk, these events contain pulses of single to a couple of photon equivalents (p. e.) and allow the calculation of the gain and other properties of the photon detectors [35]. Until version v1.0.0, the FACT-Tools analysis used the result of a single gain measurement. However, the gain is not completely stable over time. This was fixed by implementing a new service, always providing the closest available gain data for the data being analysed.

The mean arrival time of the photons is estimated by the inflection point of a third order polynomial fitted to the rising edge of the pulse. The signal of known defective pixels is interpolated from their neighbors. Figure 4.3 shows the number of photons and arrival times for each pixel of simulated proton event, while Figure 4.4 shows the same for a simulated gamma event. The third common event signature caused by a highly energetic muon reaching the telescope is shown in Figure 4.5.

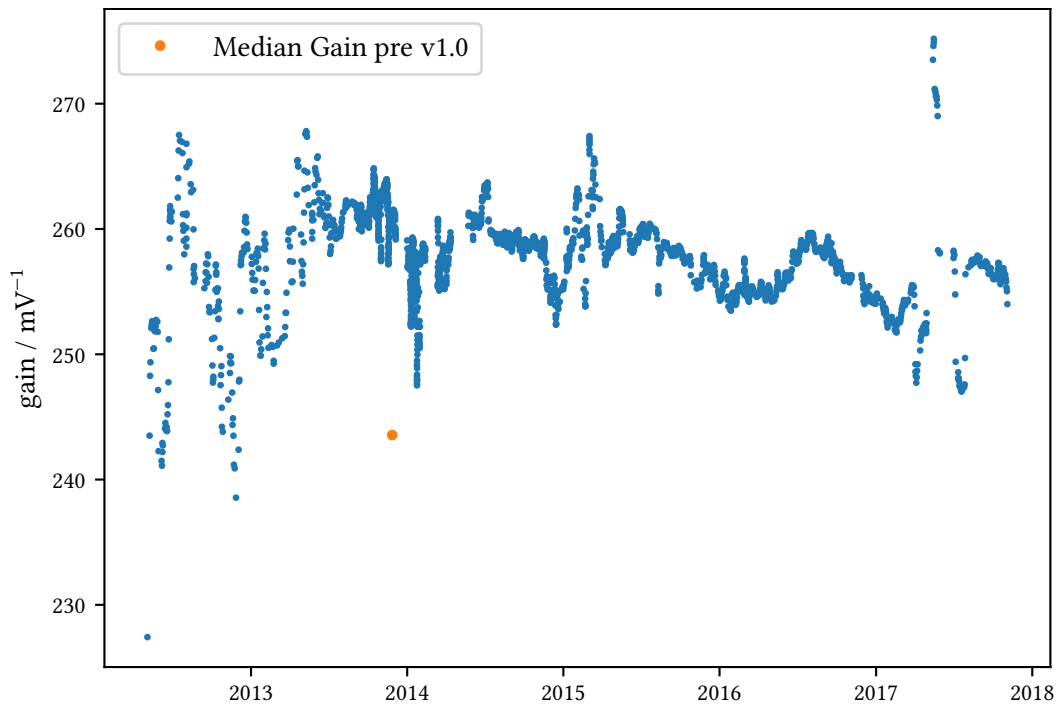


Figure 4.2: Median pixel gain over time as provided by the GainService. The file used before was from November 2013 and the contained gains are far from the mean for other dates, potentially inducing a bias in the reconstruction of the number of photons. The outliers in 2017 are from a broken electronics board in the camera, which at that time had been replaced but not yet calibrated.

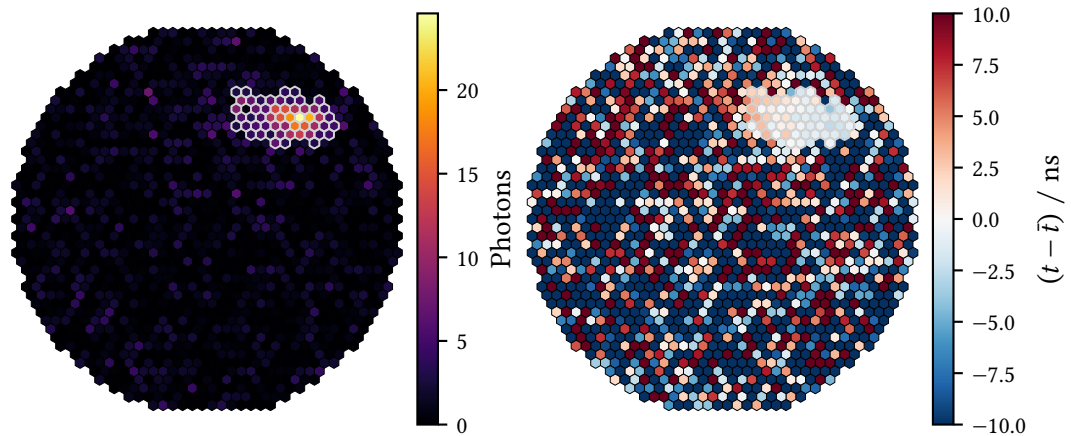


Figure 4.3: Number of photons (left) and arrival time (right) relative to the mean arrival time of the pixels selected by the image cleaning (gray border) for a simulated 15 TeV proton shower.

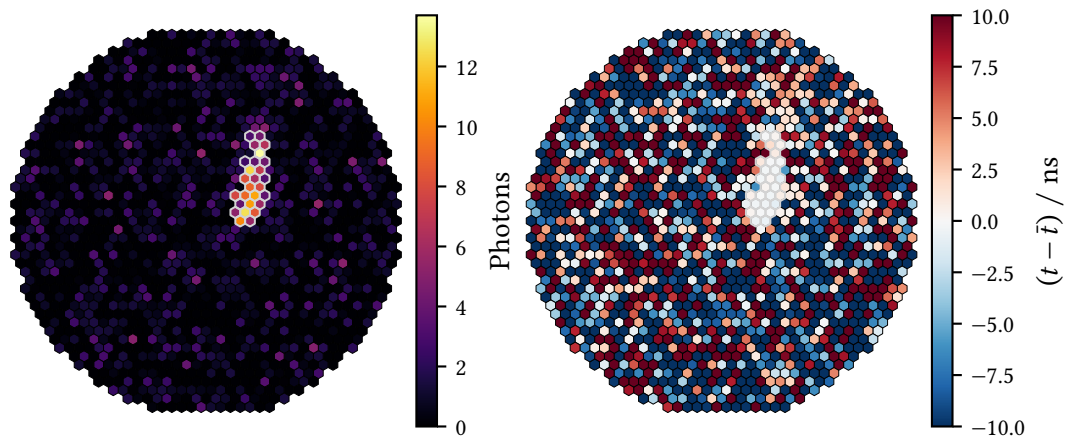


Figure 4.4: Number of photons (left) and arrival time (right) relative to the mean arrival time of the pixels selected by the image cleaning (gray border) for a simulated gamma shower with $E = 1.7$ TeV.

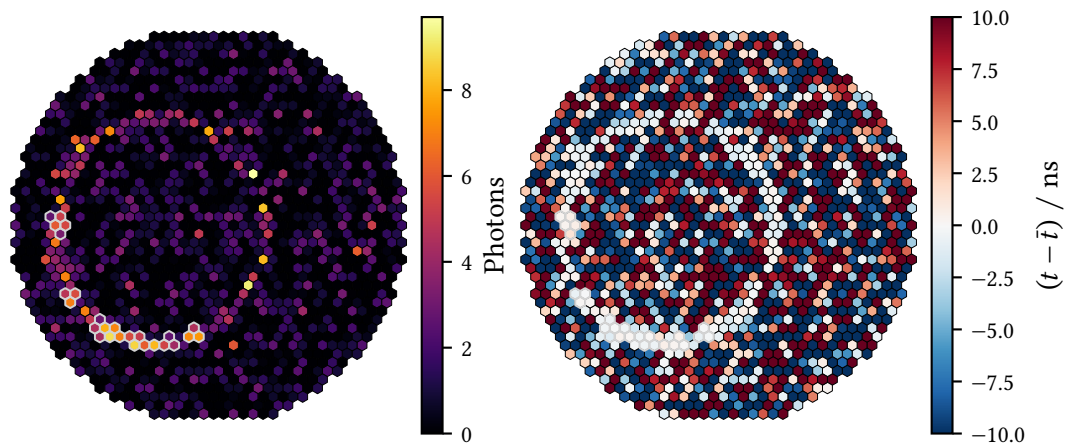


Figure 4.5: Number of photons (left) and arrival time (right) relative to the mean arrival time of the pixels selected by the image cleaning (gray border) for a simulated muon with $E = 100$ GeV reaching the telescope.

4.5 Image Cleaning

For most of the events, only a small fraction of the pixels will contain any signal created by Cherenkov photons. Goal of the image cleaning is to discard pixels that only contain noise or NSB photons.

In FACT-Tools, this is done in a stepwise procedure:

1. Find pixels containing more photons than an upper threshold t_1
2. Remove pixels with less than N neighbors above that threshold
3. Add neighbors of the remaining pixels that are above a lower threshold t_2
4. Remove pixels that have less than N neighbors with an arrival time inside a time limit Δt of the pixels own arrival time
5. Remove single pixels with less than N neighbors in the remaining pixels
6. Remove pixels that have less than N neighbors with an arrival time inside Δt
7. Remove pixel groups that only comprise pixels close to a bright star in the field of view

The last step is necessary, as a bright star in the field of view locally creates a higher noise level, which would worsen the reconstruction especially of small showers. This happens regularly for galactic sources. For Crab Nebula observations, ζ Tauri, a star with an apparent magnitude of 3.0, is inside the field of view. Before FACT-Tools version v1.1, stars had to be defined in the steering xml file. To further improve the automatic analysis, for example in case of follow-up observations for external alerts [36], a new service was implemented that automatically queries the “Yale Bright Star Catalog” [92] for the current pointing position of the telescope and identifies all stars brighter than a certain magnitude inside the field of view. For the standard processing, the maximum magnitude is 4. The noise level created by bright stars is visualized in Appendix 8. Figure 4.3 shows the pixels selected by the image cleaning for $t_1 = 5$, $t_2 = 2.5$, $N = 2$ and $\Delta t = 5$ ns. These are the values used to process all datasets in this thesis.

4.6 Image Parameterization

To further reduce the amount of data, the events are now processed from the cleaned images of amplitude and arrival time to a much lower number of features per event. It is crucial to retain as much information as possible through finding descriptive features. Much work has been put into finding such features over the last decades, a first set of features was proposed by Michael Hillas for the analysis of Whipple observations in 1985 [90], these features are still used today and are visualized in Figure 4.6. The total number of photons in the pixels after cleaning is referred to as *size*. The location of the shower image is described by the two Cartesian coordinates of its center of gravity, the average of the pixel coordinates weighted with the number of photons in each pixel. Now, to characterize the extension and orientation of the shower, a principal component analysis of the two-dimensional light distribution is performed. The covariance matrix of the pixel coordinates weighted with the number of photons in the pixels is calculated, then its eigenvalues and eigenvectors. The eigenvalues correspond to the variances along the principal components,

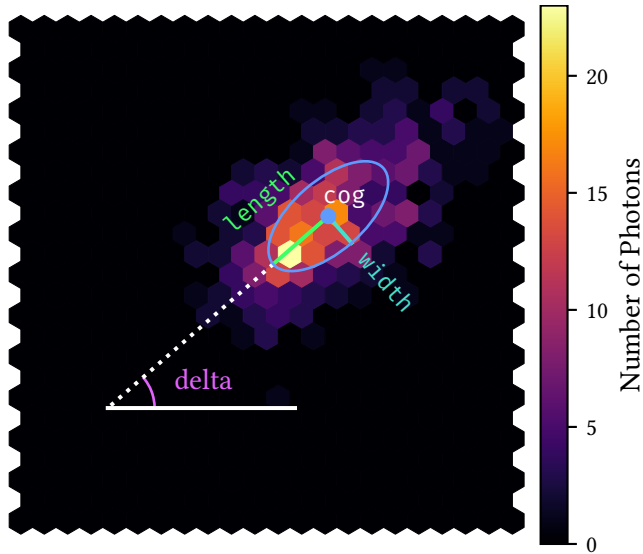


Figure 4.6: The classical Hillas features for an example shower image. Basis is a principal component analysis of the light distribution. The first component is called length and the second width. Orientation is defined by the angle of the first component to the x -Axis. For visualization, an ellipse with length and width as semi-major and semi-minor axis is shown. Higher order moments are calculated in the rotated coordinate system defined by the principal components.

the larger corresponding standard deviation is called length, the smaller width. The orientation is described by the angle of the larger component's eigenvector to the x -axis, called delta. The orientation of the large component corresponds to the shower axis, the elongation of the primary particle's trajectory. The principal components and the center of gravity define the shifted and rotated shower coordinate system, where coordinates along the major axis are called longitudinal and transversal along the minor axis. In addition to width and length, also the higher moments skewness and kurtosis are calculated along the principal components. Skewness along the shower axis can indicate the direction the shower came from. General descriptive statistics for the light distribution and the arrival times are also calculated.

Four similar features describe the amount of light in the brighter sections of the image compared to the total amount of light: `concentration_core` is the percentage of light inside the Hillas ellipse, `concentration_cog` the percentage of light of the three pixels closest to the center of gravity and `concentration_one_pixel` and `concentration_two_pixel` are the percentages of the brightest and the two brightest pixels, respectively.

As not all showers, especially at higher energies, are completely enclosed by the camera, features quantifying containment are needed. For this, the percentage of light in the border pixels of the camera can be used. This is called `leakage1` for the outermost ring of pixels and `leakage2` for the two outermost rings of pixels. The number of pixels remaining after cleaning is called `num_pixel_in_shower` and the amount of groups of connected pixels after the cleaning is called `num_islands`.

To describe the temporal development of the shower along its axis, a linear regression of the arrival times vs. the longitudinal position is performed. The resulting slope helps in determining the direction of the shower and is called `time_gradient_slope_long`.

In older versions, FACT-Tools also calculated a number of features that depended on a known point source position. It was decided to drop these features to make the image parameterization data level agnostic of any assumptions about a source. Without assumptions about a source position baked into the reconstruction, later stages of the analysis can perform the appropriate

step for different kinds of observations.

4.7 Output file format

Until version `v0.16.0`, `FACT-Tools` only supported text-based output for analysis results, namely comma separated values and JavaScript Object Notation (JSON). This had a number of disadvantages, in particular file size, writing and parsing speed and the limited amount of metadata supported by these formats. JSON while able to represent nested data well, is missing the possibility to store infinity and not-a-number values as occur often in scientific computing. To improve the situation, a new `FITSWriter` was implemented to store analysis results into FITS binary table extensions. FITS files have headers for metadata, e. g. the software version and processing date used to perform the analysis are stored. Binary tables store data more efficient, are faster to write and read. Additional header fields can be added by the process, as of `v1.1.2` the standard analysis stores information about the observed source and the Domino Ring Sampler (DRS) calibration file used.

4.8 Celestial Coordinate Systems and Transformations

Because Earth is not a stable observation platform in the universe, it is necessary to assign fixed coordinates to sources that can then be transformed into a local coordinate frame for an observer at a specific location on Earth at a given time. This section is a stark simplification of the actual complexity of this topic, which is needed to achieve the precisions at the milliarcsecond level. For an in-depth treatment, refer to [151]. (Un-)Fortunately, Cherenkov astronomy has several limiting factors that make this level of precision unnecessary. For example, the best reconstruction algorithms for stereoscopic telescopes only reach uncertainties on the arcminute level [110] at the highest energies. This will probably be irreducible due to the stochastic nature of the particle cascades in the atmosphere, limited knowledge about atmospheric conditions and deflection of the charged secondaries in the Earth magnetic field. Theoretical limits to the reconstruction of gamma rays observed with IACTs are discussed in [93]. Correspondingly, pixels of Cherenkov telescopes usually have a field of view of several arcminutes.

4.8.1 Equatorial Coordinates in the ICRS

The current standard for celestial coordinates is the International Celestial Reference System (ICRS), with the origin at the solar system's barycenter and the coordinate axes defined by catalogs of specific sources with fixed coordinates called reference frames. It is the first reference system, that does not depend on Earth's orientation at a specific point in time, called an epoch. However, the axes of the ICRS are closely² aligned to that of the equatorial system of epoch J2000.0, January 1st, 2000 at 12:00 terrestrial time [151, Chapter 4]. This achieves backwards compatibility to catalogs expressed in the formerly used reference system.

The z-axis points closely along the rotation axis of the Earth and the x-axis points towards the vernal equinox. The equinoxes are the points on Earth's orbit around the sun where the plane of

²the total deviation is around 0.02'' [151, page 105]

Earth’s equator crosses the ecliptic, the plane in which Earth rotates around the sun. Vernal refers to the equinox Earth reaches at the beginning of spring and the autumnal equinox is reached in fall. Usually, spherical coordinates are used, where longitude is called *right ascension* α and the latitude is called *declination* δ . As right ascension is related to Earth’s rotation, it is often expressed in hour angles, with $24^{\text{h}} = 360^\circ$.

Reference frames are defined by specifying the coordinates of bright extragalactic objects, mostly quasars. Quasars are a broad category of AGN with very bright accretion disks, which belong to the most luminous objects in the universe. Currently, the most precise reference frame—the third international reference frame—is made up of 4536 extragalactic radio sources observed using VLBI reaching a precision down to $30 \mu\text{as}$ [96].

4.8.2 Earth Locations

Locations on Earth are expressed using geodetic latitude λ , longitude ϕ and height above a reference ellipsoid. Nowadays it is implemented as the WGS84 reference frame by the GPS satellite network and reference stations aligned with the coordinate grid [151, Chapter 5]. In this frame, the position of FACT is $28^\circ 45' 41''$ N and $17^\circ 53' 27''$ W.

4.8.3 Local Coordinates

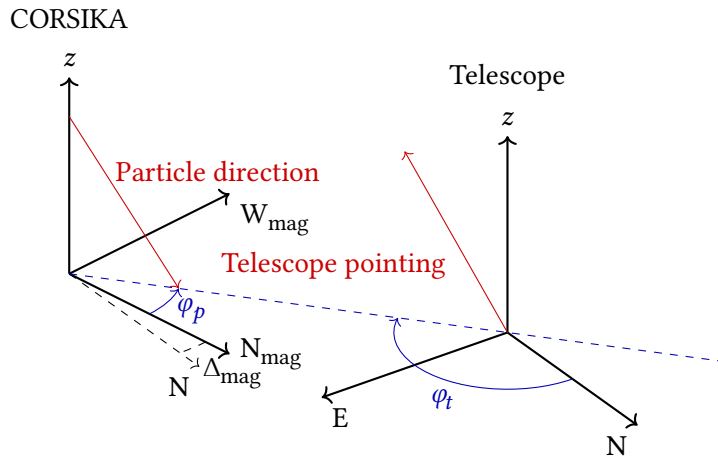


Figure 4.7: Relation of CORSIKA’s horizontal coordinate system for particle movement to the one used for the pointing direction of an IACT. $\varphi_t = 180^\circ - \varphi_p + \Delta_{\text{mag}}$, where φ_t is the pointing azimuth of the telescope in the classical horizon system, φ_p is the azimuth of the particle direction in CORSIKA’s system and Δ_{mag} is the angle between the North Magnetic Pole and the Geographic North Pole, called magnetic declination. For this visualization, $\varphi_p = 30^\circ$, $\Delta_{\text{mag}} = -10^\circ$ and thus $\varphi_t = 140^\circ$.

Local coordinates for an observer are usually expressed using a horizontal spherical coordinate system with altitude angle a as latitude perpendicular to the horizon and the azimuthal angle φ as longitude in the horizontal plane. Especially in Cherenkov astronomy, the zenith distance $\vartheta = 90^\circ - a$ is used regularly instead of altitude a . Several definitions exist for the origin and

4 From Raw Data to Image Parameters

direction of φ , the most common being $\varphi = 0^\circ$ for the direction of geographic north, going positive to the east ($\varphi = 90^\circ$). CORSIKA uses the direction towards the North Magnetic Pole as origin and grows positive towards west. This is used to simplify calculations for the deflection in Earth's magnetic field. A downside of this approach is the introduction of a time and location dependent offset in azimuth of CORSIKA's coordinate system to the ones used in astronomy³. The position of the North Magnetic Pole is not identical to the Geographic North Pole and changes over time. CORSIKA uses this coordinate system to specify the movement direction of the particle, which is directly opposite to the pointing direction of a telescope observing said particle, this situation is shown in [Figure 4.7](#). `ceres` converts from particle direction to pointing direction when reading CORSIKA output and corrects for the difference between magnetic and geographic north.

4.8.4 Detector Coordinates

FACT uses a detector coordinate frame that uses two dimensional Cartesian coordinates in the focal plane of the telescope. The coordinate axes are defined so that when standing in the reflector dish, looking onto the camera, x points right and y points up. This coordinate frame is used to calculate image parameters and is used to express reconstructed source positions before transforming them into the horizontal coordinate frame or ICRS.

4.8.5 Time

The current time system based on Earth's orientation towards the sun is called Universal Time 1 (UT1) and is measured by VLBI observations of distant sources, and is thus only available after some time, usually observations are performed and values are published daily. The duration of a UT1 day underlies a general slow increase and random fluctuations. Civil time keeping is using Coordinated Universal Time⁴ (UTC), a timescale based on Temps Atomique International (TAI), the time kept by hundreds of atomic clocks around the globe monotonically counting SI seconds. UTC is kept within 0.9 s of UT1 by introducing leap seconds up to twice a year. At time of writing, UTC is 37 seconds behind TAI. When precision at the arcsecond level or below is not required, directly using UTC for the Earth rotation angle is a good enough approximation. Otherwise, tabulated values of the difference between UT1 and UTC have to be used. [151, Chapter 3]

4.8.6 Implementation of Coordinates and System Transforms

In general, any conversion from the celestial reference frame into the local frame of an observer, e. g. into the horizon system, requires taking into account precession and nutation of Earth's rotation axis, Earth's rotation, movement of the pole relative to Earth and the position of the observer. Precession is the longterm circular movement of the rotation axis due to the pull of the sun, moon and planets. Nutation is the additional, periodic deviation from this. Unfortunately, no library for java was readily available to perform these transformations. Before version `v1.0.0`, FACT-Tools only had a very rudimentary support for transforming from equatorial coordinates into camera coordinates, completely disregarding precession and nutation that was thus only

³This the same effect that needs to be compensated for classical magnetic compass navigation.

⁴This abbreviation was chosen as a compromise between French (`temps universel coordonné`) and English that would not favor one language.

accurate to a level of 10' steadily getting worse with time. Because pointing and source position were transformed in the same way, the relative positions in detector coordinates were still correct. For v1.0.0, a full system of coordinate transforms implementing equatorial, horizontal, detector and earth location frames and transformation between these systems has been developed. The transformation between equatorial and horizon system uses an approximation for precession resulting in an accuracy better than 30'', see Figure 4.8 [72].

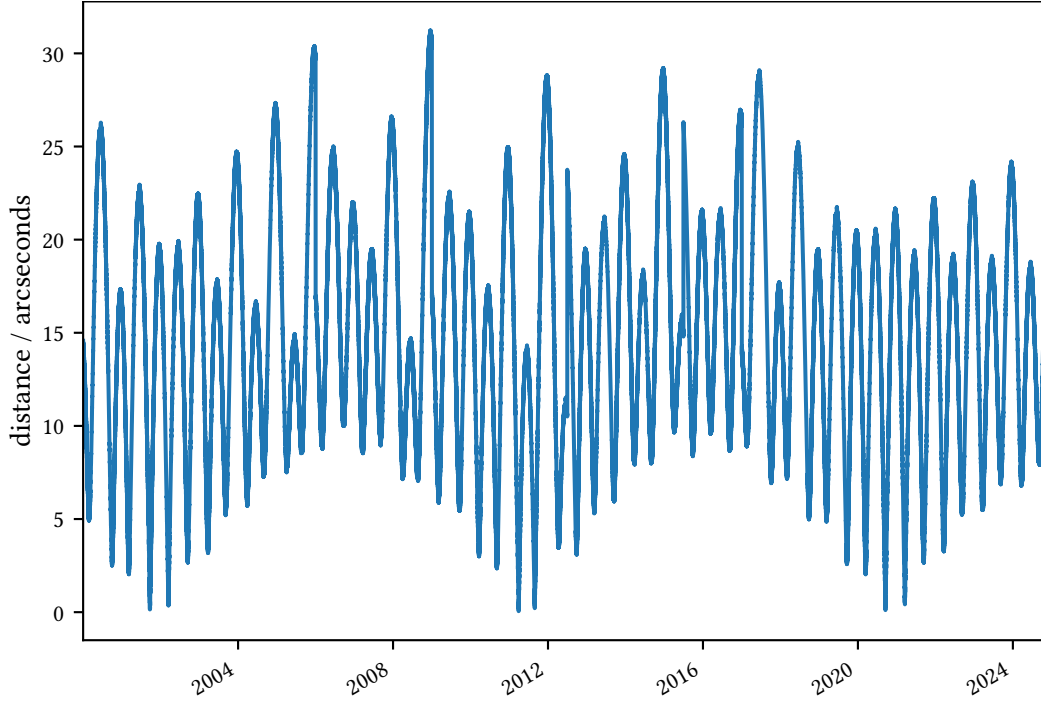


Figure 4.8: Deviation between FACT-Tools and astropy [136] for coordinate transforms from equatorial to horizontal of the Crab Nebula over 25 years in 6 hour intervals. This basically shows the effect of Earth's nutation, which is not taken into account in the coordinate transforms in FACT-Tools.

To convert from horizontal coordinates to camera coordinates, the spherical coordinates of the direction of interest is converted into Cartesian coordinates, where x points north, y points east and z points up. To align the coordinates with the focal plane, the frame is rotated by the azimuth of the pointing direction φ_p around the z -axis and then by the zenith angle of the pointing direction θ_p around the y -axis. This is equivalent to rotating the vector r by the negative angles:

$$\mathbf{r}' = \mathbf{R}_y(-\theta_p) \cdot \mathbf{R}_z(-\varphi_p) \cdot \mathbf{r} \quad (4.1)$$

The coordinates are then projected down onto the focal plane, which is in a distance of the focal length f from the origin in the reflector dish. From similar triangles:

$$x'' = \frac{x'}{z'} \cdot f \quad y'' = \frac{y'}{z'} \cdot f \quad (4.2)$$

with $\mathbf{r}' = (x', y', z')^T$. Now the reflection of the mirror is applied, and the coordinate axis are

identified with the orientation as wished from the definition of the camera coordinate system. This finally yields:

$$c_x = -y'' \quad c_y = x'' \quad (4.3)$$

The transformations are visualized in Figure 4.9.

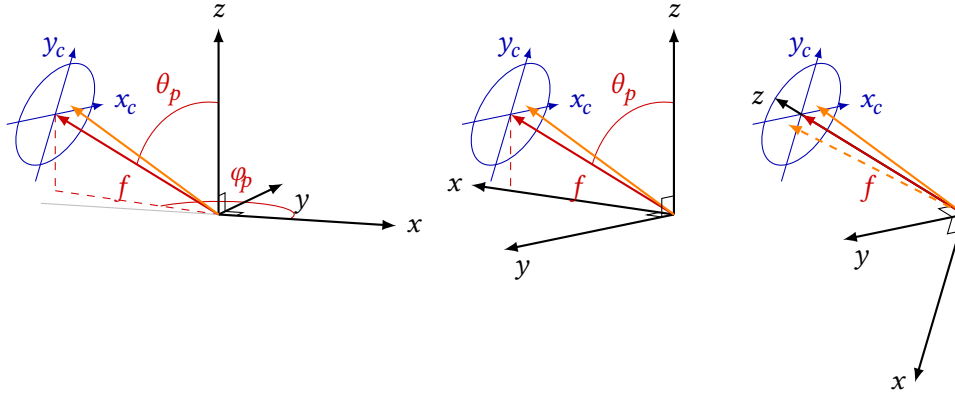


Figure 4.9: From left to right: initial situation, coordinate frames after applying the rotation in azimuth, coordinate frames after applying the rotation in zenith. The dashed orange line shows the source vector after applying the reflection at the mirror.

4.9 Processing of all FACT data to Image Parameter Level

In total, FACT took approximately 645 TB of compressed raw data until December 2019. All of FACT’s raw data is stored in a tape archive at the Integral Science Data Center (ISDC), a backup exists on a tape archive in Würzburg. Only a subset is available for processing on hard disks at the LESTA cluster due to space constraints.

To analyze all this data using FACT-Tools for image parameter extraction, a tool for automatically processing FACT data runs was implemented into the `erna` [127] package. The general structure is very similar to `mopro3` described in section 3.4. Again, a database is used for job management. The needed information about FACT runs and calibration files are duplicated from FACT’s main database into the `RawDataFile` and `DrsFile` tables. The `xml` steering files and the `FACT-Tools-jars` are also stored in the database and are automatically copied to the executing node when needed. A single `FACT-Tools` job is the application of a certain `FACT-Tools` jar in combination with an `xml` file, a raw data file and a `DRS` calibration file. Like with `mopro`, additional information needed for the job processing is also stored in the database, this includes output file path, processing status, walltime for the cluster job and a hash of the result file to be able to detect data corruption. Unlike the production of simulated data, the processing of the observed data depends on external files. Thus, another error state is possible: input file missing. The `submitter` program runs perpetually on the computing node able to create `SLURM` jobs for the LESTA cluster at ISDC, continuously checking the database for new jobs to process. If such a job is found, it is checked if all needed input files are present at the configured location. If yes, the job is submitted into the cluster, If not, the job is filed as “input file missing”. Another program runs once a day, checking if missing files have been

4.9 Processing of all FACT data to Image Parameter Level

restored from tapes to disk and if that is the case, the jobs are reset and can be processed by the submitter.

The FACT-Tools standard analysis has been applied to all FACT data measured between August 21st, 2012, the date where the currently used format for the auxiliary data was introduced, and June 30th, 2018. In total, 149 478 runs were processed. This was achieved by restoring the data in monthly chunks from the tape archive to the distributed file system of the LESTA cluster. Restoring the data from tape to disk took much longer than the actual processing. In total, processing of all available FACT data took three months, but happened fully automatically. All these runs are now available at the image parameter level in standard conform FITS files for further analysis.

Reconstruction of Particle Properties using Machine Learning

5

The final tasks of the event-wise analysis of IACT data is the estimation of the physical properties of the measured events. Three properties must be reconstructed to be able to perform physical analysis of gamma-ray sources, e. g. measuring their spectral energy distribution:

Particle type As described in [section 1.5](#), IACTs measure thousands of air showers induced by charged cosmic rays for each measured gamma ray. To reduce this background as effectively as possible, events are classified as either gamma-ray or hadron induced. This is a classification task.

Primary energy While techniques like unfolding could directly estimate a spectral distribution from several input features, condition of the inverse problem usually improves when using a direct estimator of the primary energy. This is a one-dimensional regression task.

Origin The direction of origin also plays a crucial role in the background suppression, as the cosmic ray background is isotropic compared to the compact or point-like gamma-ray sources. The two-dimensional regression to estimate the origin of a gamma ray is especially hard for monoscopic IACTs, as multiple telescopes allow simpler and more precise geometric reconstruction methods.

Historically, these tasks were performed by applying event selection criteria, often called *cuts*, for the particle classification and hand-crafted regression formulas fitted to simulations for the energy estimation and reconstruction of origin. Modern methods of machine learning have improved performance and removed human biases from these steps.

While there have been first successes [[141](#), [120](#)] in using convolutional neural networks to directly reconstruct these properties from lower stages of the event processing¹, these analyses are all in a prototype stage and seem much more dependent on mismatches between simulations and observations than classical machine learning approaches using the image parameters as input. This is why the focus here will be laid to supervised, decision tree based ensemble methods.

For a long time, there has been no common file format or other representation of reconstructed gamma-ray events, which could be used to enable analyses combining multiple observatories. Each collaboration used their own file format and conventions. An effort to standardize data formats for gamma-ray astronomy, mainly in preparation for CTA—but also to enable joint analyses of the currently operating telescopes—was started in 2015. A first version of the specification based on the FITS standard has been published in 2016. [[58](#), [59](#)]. This is an ongoing effort and part of this work was to enable the storage of FACT reconstructed event lists in the new, open format.

¹like the number of photons per pixel and their arrival time or even from the voltage time series in each pixel

5.1 Supervised Machine Learning

Machine learning is the process of finding an estimator f for a quantity Y given some inputs X by using generic classes of estimators f and fitting the parameters of these estimators by minimizing a loss-function.

In the following, I will use the notation which closely resembles the one used in [83]: The target property Y of several observations will be represented by the vector y . The p input quantities X_j for these N observations will be stored in the $N \times p$ Matrix X , where X_j denotes the j -th column, all observations for input quantity X_j and X_i denotes the i -th row, all input quantities for observation i . For regression problems, where a continuous quantity is estimated, Y is directly the target variable. For classification problems, where observations are to be sorted into different groups, a target variable $\hat{Y}_g \in [0, 1]$ for each class g is estimated and a sample is assigned class \hat{C}_g if \hat{Y}_g is greater than a prediction threshold t_g . For binary classification, this simplifies to just predicting a single \hat{Y} and using a single *prediction threshold* t splitting the samples into two classes.

In supervised machine learning, the estimator's parameters are optimized on a dataset where the target variable Y is known. This is commonly called *training*. A trained estimator can then be used to predict \hat{Y} for observations with unknown Y .

An important step is the validation of trained models on datasets that were not used for the training but have known Y to calculate quality metrics that compare the estimated quantity \hat{y} with the true quantity y . This is needed to prevent *overfitting*, the effect that a model learns the training dataset "by heart" instead of the underlying structure and thus fails to generalize to data it has not been trained on. Metrics and validation will be discussed in [subsection 5.1.3](#).

Many different algorithms exist for both regression and classification tasks, that differ greatly in their assumptions about the data, their generality, the loss-function they try to minimize. Most algorithms are also tunable with so-called hyperparameters. These are properties of the models that stay fixed for a single training step but have to be optimized for the current application. A simple example for a hyperparameter is the number of next neighbors k considered in a k -Nearest-Neighbor classifier [83, chapter 9.1]. This all can greatly influence their quality metrics on a specific problem. In the following, I will introduce decision trees and random forests. Random forests have proven to be a very general, robust, fast to train and fast to apply model. While boosting methods can often achieve slightly better performance, training time is usually much higher as their training cannot be parallelized.

5.1.1 Decision Trees

Decision trees locally optimize a loss function by recursively splitting the feature space into subregions. The most common case are binary trees, that in each step split each subregion into two new regions. [41] Finding the optimal subdivision of a feature space using binary splits is NP complete [105], so instead of finding *the optimal* solution, a greedy algorithm to find *a good* solution is employed: Starting from the full feature space R , the best split of all possible splits in all p features is done, splitting into two subspaces R_1 and R_2 . This procedure is repeated recursively in all resulting subspaces until a stopping criterion is met, e. g. a maximum number of splits or a minimum number of samples in the resulting subspaces. Each split is called a node, each final subspace is called a leaf. For a regression tree, the output is the average of the training data in

the leaf, for classification, either the majority class or the partitions of the classes in the leaf as continuous score $\in [0, 1]$ are returned. The maximum number of splits is called the depth of the tree and can be used to constrain the model complexity and thus overfitting.

For classification, there are two commonly used criteria to determine the best split, cross-entropy and the Gini-Index, which are pretty similar, for two classes with p being the proportion of one class, these are:

$$\text{Gini Index} = 2p(1 - p) \quad (5.1)$$

$$\text{Cross-entropy} = -p \log p - (1 - p) \log(1 - p) \quad (5.2)$$

For regression, the mean squared error

$$\text{mse} = \sum_{X_{ij} \in R_1} (y_i - \hat{c}_1)^2 + \sum_{X_{ij} \in R_2} (y_i - \hat{c}_2)^2 \quad (5.3)$$

is the most common loss function. The criterion is evaluated for all possible splits in all features X_j to find the best possible split for the current node. [83, chapter 9.2]

5.1.2 Random Forests

Because of the greedy algorithm, decision trees have intrinsically a high variance. Small fluctuations in the training dataset can alter the very first splits and thus alter the full tree. To achieve a better and adjustable bias-variance-trade-off, Leo Breiman introduced random forests [43]. A common approach to reduce variance is *bagging* [42]. Multiple models are trained on slightly different datasets created by sampling with replacement from the original training dataset creating an ensemble. For the final output, the outputs of the single models are averaged. Random forests are a bagged ensemble of decision trees with an additional randomness to further diversify the single decision trees: only a random subsample of the features is considered for the best split in each node.

5.1.3 Quality Metrics and Validation

There are two different quality metrics relevant for the evaluation of the models trained to perform IACT event reconstruction. In this section, I will focus on quality metrics commonly used in statistics and machine learning for the evaluation of classifiers and regressors. The quality metrics commonly used in gamma-ray astronomy to describe the performance of the full analysis chain of an IACT will be discussed in chapter 6. All quality metrics for binary classification build upon the confusion matrix, a visual explanation of which is shown in Figure 5.1. The most common quality metrics for binary classification are:

Precision also referred to as purity or positive predictive value, is the percentage of positive samples in the selected samples:

$$\text{precision} = \frac{\text{tp}}{\text{tp} + \text{fp}}. \quad (5.5)$$

Recall also known as true positive rate, efficiency or sensitivity, is the fraction of selected positive samples to all positive samples:

$$\text{recall} = \frac{\text{tp}}{\text{tp} + \text{fn}}. \quad (5.6)$$

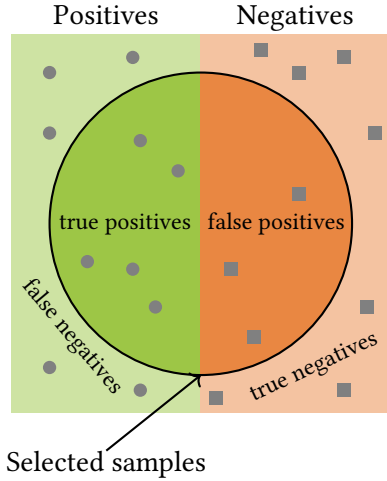


Figure 5.1: Confusion for a binary classification problem. On the left are all samples belonging to the positive class, on the right the negative ones. The circle encompasses all samples classified as positive. This leaves us with four important areas, which are the entries in the confusion matrix. Correctly classified positive samples are the true positives tp , correctly classified negative samples are called true negative tn . The falsely classified samples are called appropriately, short fn and fp . The confusion matrix is then

$$\begin{pmatrix} tp & fp \\ fn & tn \end{pmatrix}. \quad (5.4)$$

Adapted from [135].

Accuracy is the fraction of correctly classified samples

$$\text{accuracy} = \frac{tp + tn}{tp + tn + fp + fn} \quad (5.7)$$

F-Score is the harmonic mean of precision and recall, where the weight β can be used to put more emphasize on recall ($\beta < 1$) or precision ($\beta > 1$):

$$f_\beta = (1 + \beta^2) \frac{\text{precision} \cdot \text{recall}}{\beta^2 \text{precision} + \text{recall}} \quad (5.8)$$

All these metrics require the classification into the classes \hat{G} . As most models provide a score \hat{Y} , it is often useful to optimize model hyperparameters on a metric that uses \hat{Y} , so that the optimization of model hyperparameters is independent of the optimization of the prediction threshold t_p . In the ROC curve [70], the true positive rate (tpr) is plotted against the false positive rate (fpr) for all possible thresholds t_p . The area under this ROC curve A_{roc} is a metric that describes classifier performance independent of t_p with a single number and is thus a common metric for the optimization of hyperparameters. Three prototypical cases for ROC curves are shown in Figure 5.2. The ROC curve is also independent of the class balance.

For regression problems, the mean squared error is a commonly used metric, but it is a highly problem specific one. A generalization is the *coefficient of determination* or r^2 -score, which is defined as

$$r^2 = 1 - \frac{\sum_{i=0}^N (y_i - \hat{y}_i)^2}{\sum_{i=0}^N (y_i - \bar{y})^2}, \quad (5.9)$$

where \bar{y} is the arithmetic mean of all y_i . This metric is 1 for the perfect regressor, 0 for a regressor always predicting \bar{y} and < 0 for any regressor doing worse than simply predicting the mean.

All metrics have limited informative value without considering how they depend on the statistical fluctuations of training and application dataset. A common method to not jeopardize the available data for training too much by splitting into independent training and test sets, is cross validation. Here, a dataset is split into N parts but training happens N times on $N - 1$ parts and testing and metric evaluation is done on the remaining part. Thus all test parts are independent, but the

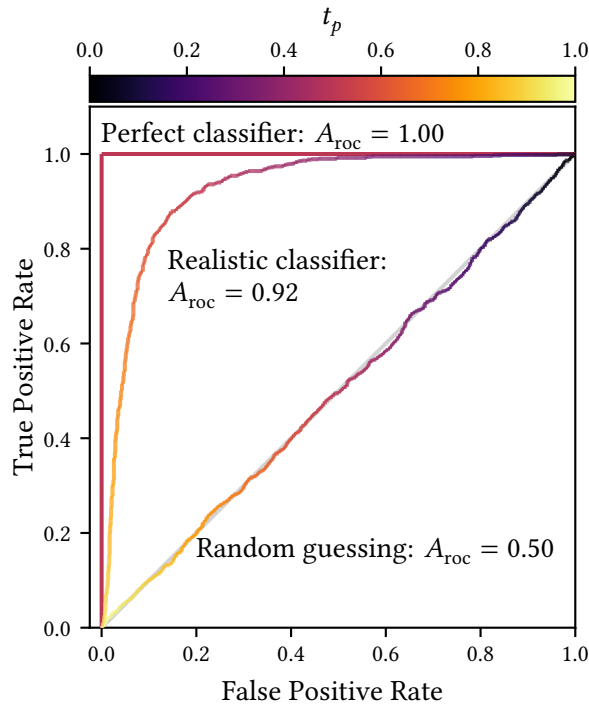


Figure 5.2: Receiver Operating Characteristic (ROC) curves for three different classifiers for a simple toy example. The perfect classifier creates a square, as $tpr = 1, fpr = 0 \forall t_p > 0$ and has $A_{roc} = 1$. Randomly guessing yields the diagonal and thus $A_{roc} = 0.5$. Any classifier performing better than guessing \hat{Y} uniformly will have $0.5 < A_{roc} < 1$.

training data is overlapping and each sample in the training data is used for evaluation exactly once.

5.1.4 The aict-tools

To be able to train, validate and apply models using supervised machine learning to IACT data, the `aict-tools` [125] have been developed. This python package provides command line utilities to train and apply models from the `scikit-learn` [132] library to image parameters of IACT events stored in columnar oriented `hdf5` [75] files.

Models provided by `scikit-learn` include linear regression, naive Bayes, support vector machines, tree based algorithms and ensemble learners, for example random forests and several versions of boosted decision trees.

The models are configured using `yaml`-Files [29], a human and machine readable, plain-text data format. An example configuration for an energy estimation using a linear regression of five features is shown in Listing 5.1.

Models are serialized for later application using either `joblib` [152], which is an efficient storage format for Python classes containing large arrays, or one of two language independent, standardized formats for machine learning models: Predictive Model Markup Language (PMML) [82] or the ML extension for classical machine learning models of the Open Neural Network Exchange Format (ONNX) [129]. This can be used to apply models trained using `aict-tools` using other programming languages and tools, for example in `FACT-Tools` [46].

Three pairs of `aict_train_` and `aict_test_` programs are provided for the particle classification, the energy estimation and the reconstruction of origin using the `disp` method, respectively.

Listing 5.1: Example `aict-tools` configuration for an energy estimation using `scikit-learn`'s linear regression.

```
seed: 0 # seed for the random number generators

energy:
  regressor : linear_model.LinearRegression()

  n_signal: 50000
  n_cross_validations: 10

  target_column: corsika_event_header_total_energy
  output_name: gamma_energy
  log_target: True

features:
  - size
  - length
  - width
  - leakage1
  - leakage2
```

The project was started by Kai Brügge in 2016 and I joined in early 2017, when only energy estimation and particle classification were implemented. The main focus at the time was to replace the RapidMiner [116] based analysis developed in [149], with some key points in mind:

- Better reproducibility through simple configuration files in text format compared to the GUI based RapidMiner approach.
- Better performance for application to larger datasets. Training and application times of the Weka random forest in RapidMiner were much larger than for the `scikit-learn` random forest.
- No dependencies on closed source, proprietary software².

Later, as one major part of the work done for this thesis, I developed and implemented the origin reconstruction into the `aict-tools` based on the `disp` method [107] which is explained in detail in section 5.4.

In addition to the tools for model training and application, the `aict-tools` also provide utilities to

- apply event selection cuts, also configured via a `yaml` file,
- split datasets into multiple subsets, e. g. for training and testing,
- create plots of quality metrics to evaluate model performance.

²While RapidMiner is open core, the free version has a limitation of only 10 000 rows for datasets and can also only utilize one CPU core.

5.2 Particle Classification

Particle classification is done using a random forest classifier trained to distinguish gamma-ray induced showers (signal) from proton induced showers (background).

The output is called `gamma_prediction`, with values close to 1 indicating a shower likely induced by a gamma ray. Compared to the previous, RapidMiner based analysis in [149], all features depending on the known position of a point source were excluded from the training because of three reasons:

- It greatly simplifies application of the models as they only have to be evaluated once and not for each of the on and off positions using the feature set for the corresponding position.
- It enables predicting a single reconstructed position for each event on the sky completely independent of any assumptions about any source.
- It separates the two tasks of estimation of origin and particle classification. In [149], because the distance to the assumed source position was used for the classification, these steps were intertwined.

The second reason is important for the creation of skymaps or in cases where no source position is known precisely or at all, e. g. when FACT performs follow-up observations of alerts by the IceCube experiment, where the neutrino direction typically has uncertainties in the order of 1° [23]. This is also the reason, why training is now done using diffuse gamma rays instead of gamma rays observed in wobble mode. Using diffuse gamma rays, the model should be able to better generalize in separating gammas from protons, and not just selecting events close to the wobble radius.

5.3 Energy Estimation

Energy estimation is done using a random forest regressor trained to predict the natural logarithm of the energy with the mean squared error (MSE) as loss function. The choice of using the logarithm as opposed to the value itself is motivated in the large domain of the primary energies. The loss function would be completely dominated by few very high energy events if evaluated on the values themselves. Using the logarithm of the energy, the lower energy particles also contribute to the loss and overall performance over the whole energy range is improved.

5.4 Origin Estimation

In general, the estimation of the gamma ray origin is a two dimensional regression task, as either two coordinates in the detector plane or on the sky have to be estimated. To simplify the task, the assumption is made, that the source position lies on the reconstructed shower axis. While this introduces a source of error if the shower axis is not properly reconstructed, this simplifies the task from a two dimensional regression to a one dimensional regression and a classification task. The objective of the regression task is to find the absolute distance from the center of gravity of the shower image to the origin of the shower called `disp` and the classification task is to find the direction on the shower axis, which can be interpreted as the sign of `disp`. This is also referred to as “head-tail-disambiguation” and is shown in Figure 5.3.

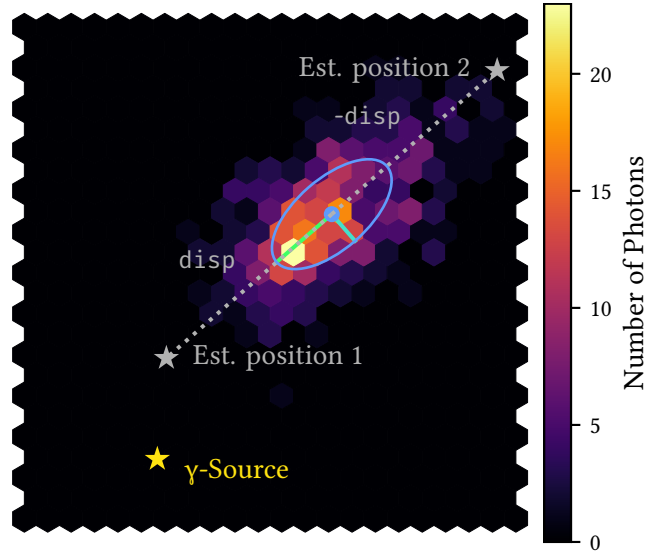


Figure 5.3: The two possible reconstructed source positions for a given $|\text{disp}|$, the selection of the correct one is the target of the head-tail disambiguation.

This method was developed for the Whipple telescope [107], greatly improving the sensitivity for point sources over the methods used before, that did not predict a source position at all but only used the orientation of the Hillas ellipse. Disp was parameterized using

$$\text{disp} = \xi \left(1 - \frac{\text{width}}{\text{length}} \right), \quad [107, (7)] \quad (5.10)$$

which is also the parameterization used by the FACT-Tools analysis. The parameter ξ was found either by optimizing it on observed data of a known, strong source as in [107] or on simulations as for the FACT-Tools. As this parameter ξ is just a fixed number, (5.10) is not able to predict $|\text{disp}|$ well over a large energy range. The value used in the FACT-Tools, $\xi = 1.38^\circ$, performed best around an energy of 3 TeV, see Figure 6.16, but much worse at higher energies. The FACT-Tools head-tail-disambiguation used features dependent on the known position of a point-source. While this greatly increased the efficiency of the prediction—especially for low energy gamma rays—it also increased the false positive rate as all showers including the diffuse hadronic background were more often predicted towards the suspected gamma source. The other drawbacks of using source-dependent image features discussed in section 5.2 also apply here. As the prediction is not independent of the source position, it has to be repeated for the off positions, increasing the computational costs with each off position. It also prevents the creation of skymaps, as there is no single prediction for an event.

The MAGIC experiment improved upon the simple parameterization in (5.10) using parameters depending on image parameters, e. g. size, to model energy dependencies and leakage to better handle showers not fully contained in the camera [61]. Finally in 2009, before MAGIC-II was built and MAGIC thus moved to stereo reconstruction techniques, using a random forest regressor to estimate $|\text{disp}|$ was evaluated and found to perform better than the previous parameterization [139]. In that case, the prediction for sgn disp was done using the difference between the image center

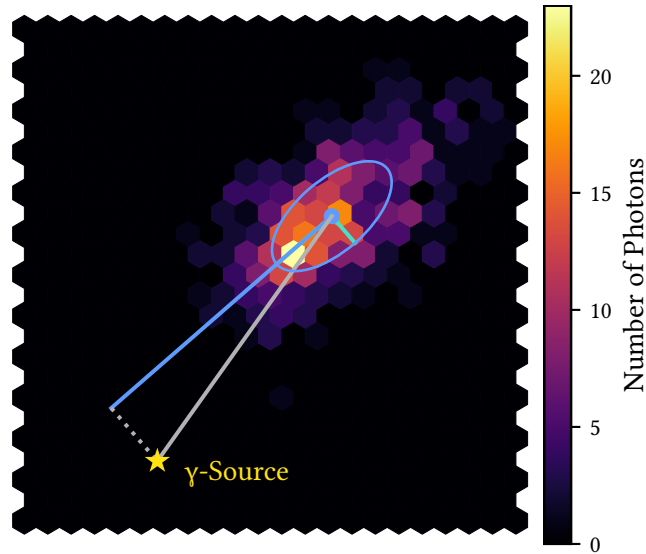


Figure 5.4: The two possibilities for choosing the label for training the disp regression. The gray line shows the distance from the cog to the source position, the blue line shows the projection onto the reconstructed shower axis.

of gravity and its brightest pixel, which is related to the skewness of the light distribution. To adapt these methods for the FACT-Tools analysis chain, the disp estimation was built into the aict-tools. To further improve over previous works, a random forest classifier is employed for the head-tail-disambiguation. This results in a single prediction of origin for each event in the detector coordinates frame, which can be transformed to a position in the ICRS as described in [section 4.8](#). The predicted source position can be used to create skymaps and is also mandatory in the open gamma-ray astronomy (OGA) fits format [58] for IACT event lists.

As the determination of the shower axis is not always perfect, there are two possibilities for the training of the disp regression. The first is to assume δ is correct and just use the distance of the source position to the center of gravity. The second is to transform the source position into the shower coordinate system and taking the coordinate along the main axis as disp. Both versions are the same if δ has no error. This is visualized in [Figure 5.4](#). It was found that the two differences do not differ in their results significantly and the first version was chosen.

Instrument Response Functions and Sensitivity of FACT

6

In this chapter, I will present the sensitivity and other performance characteristics of FACT together with the analysis methods described in Chapters 4 and 5.

First, I will present the used datasets and an event pre-selection in section 6.1, then I will discuss the instrument response functions in section 6.4. In section 6.5 the flux sensitivity will be evaluated on observed and simulated data.

6.1 Datasets

In the following, I will present the simulated and observed datasets used for estimating the sensitivity of FACT as well as the event pre-selection. Finally, I will look into the agreement between simulated and observed data.

6.1.1 Simulated datasets

Four simulated datasets are used for the analysis:

- Diffuse protons are needed for the training of the particle classification and the performance evaluation.
- Diffuse gamma rays are needed for the training of the particle classification and origin reconstruction models.
- Gamma rays from a point source observed in wobble mode are needed for the training of the energy regression, calculation of the instrument response functions and the sensitivity calculation.
- Diffuse helium nuclei are simulated as additional background class but only used for comparison of observations and simulations.

The diffuse and point-source gamma rays are generated from the same CORSIKA simulations through different settings for CERES. The run-wise settings for the corsika simulations are listed in Table 6.1. For all simulated datasets, the azimuth angle was simulated over the whole 360° range, each individual run covering 10°. The zenith angle was simulated between 0° and 30°, each run with a range of 1°.

The simulations were performed at the LiDO₃ [97] cluster at TU Dortmund using the mopro3 orchestration software described in section 3.4 and would have taken approximately 75 years on a single CPU. The output size of the event io files containing the Cherenkov photons on the ground have a total size of 6 TB for the gamma rays and 68 TB for the protons after compression using zstd.

Table 6.1: Overview of the CORSIKA runs performed for this work

Primary	E_{\min} / TeV	E_{\max} / TeV	γ	R_{\max} / m	N_{showers}	Reuse	N_{runs}
Gamma	0.2	100	-2.0	300	5000	1	10 800
Proton	0.1	200	-2.0	500	10 000	20	21 600
Helium	0.1	200	-2.0	500	5000	20	1080

The datasets simulated for [149] using `MMCS 6.5` introduced an additional photon acceptance (APA) for the detector simulation, that only accepts photons with a probability of 85 %, randomly discarding 15 % of the photons reaching the detector. This was motivated by unsimulated obstruction of light by the telescope’s structure and neglected imperfect transparency of the Winston cones. However, as described in chapter 4, the way FACT-Tools handles the gain was changed. As this also has an effect of the reconstructed amount of light, the value for this setting is also revisited. Three different sets of simulations were produced, using APA values of 85 %, 95 % and deactivating it completely (100 %). All other settings for the detector simulation stayed unchanged from [149, pages 131 sq.] The full analysis will be performed for the three different settings independently and the results will be compared.

The detector simulation for the simulated gamma rays is done twice, once to produce the diffuse gammas using a maximum scattering angle of 6° and once to produce gammas from a point source observed in wobble mode using a wobble distance of 0.6° . In total, this leads to eight simulated datasets with an overall size of 25 TB, which was processed by FACT-Tools using the `erna` package [127], also on `lido3`. `erna` allows to directly collect the output of many parallel running FACT-Tools jobs into a single `hdf5` file, as used for the further analysis. The FACT-Tools processing is by far the fastest processing step. Compared to the 75 CPU years for the CORSIKA simulations, 5.3 CPU years for the CERES simulations, running FACT-Tools only took under one CPU year.

The proton and point-source gamma-ray dataset are split into two parts, one each for training of the separation model using a quarter of the simulated events and the remaining three quarters for the performance evaluation. The diffuse gamma simulations are only used for the training of the separation and `disp` models.

If no mention of APA is made, results are shown for the dataset with $\text{APA} = 85\%$ and results for the other datasets are in Appendix 15.

6.1.2 Crab Nebula Observations

To evaluate the quality of the simulations, calculate the sensitivity on observed data and estimate the spectral energy distribution of the Crab Nebula a set of high quality runs taken between October 2013 and February 2014 with a total observation time of 91 h is used. This is the same data sample as used in [149] for direct comparison. The runs were selected using the following criteria, also from [149]:

Camera current below $8\ \mu\text{A}$. The current created by the SIPMs is a direct measure for the amount of NSB photons. This limits the sample to dark night conditions.

Moon zenith distance greater than 100° , excluding runs where the moon is above or just below the horizon.

Zenith distance of the source smaller than 30° . This eliminates the need for special treatment of zenith dependence and offers the best sensitivity.

Trigger rate between 40 and 85 Events per second. This is the nominal operating range of FACT, higher or lower values indicate bad environmental conditions like clouds (lower rate) or illumination by a light source like car lights (higher rates).

Trigger readiness during more than 95 % of the observation time, this mainly excludes runs where the strong LASER light of the MAGIC LIDAR system interfered with data taking.

Trigger threshold smaller than 350 also indicating dark nights.

The Structured Query Language (SQL) query to extract the runs from FACT's database is in Appendix 10.

A tool to collect the single FITS files created by the automatic processing described in section 4.9 provided either with a list of runs or such selection criteria into a single hdf5 file has been implemented into the `erna` package and was used to create the dataset of the Crab Nebula observations.

Raw data for a subset of this dataset containing runs worth 17.7 h of observation time was published in November 2017 by the FACT collaboration [68]. An analysis of the publicly available data based on the analysis described in this work, using the `FACT-Tools` and `aict-tools` has been published at [124]. The reconstructed event data was converted into the data format for open-gamma ray astronomy [58] by [47] and used for the first combined analysis of Crab Nebula observations by all currently operating Cherenkov telescopes in [121].

6.1.3 Event Pre-Selection

The target of the event pre-selection is to remove hard to reconstruct, not properly or not at all simulated events. The selection criteria are discussed in detail in [149, pages 56 sqq.] and are now applied using the `aict-tools` to all datasets. This pre-selection is performed before training the reconstruction models. The pre-selection reduces the amount of events for observed data roughly by a factor of two, compare Table 6.2, and dramatically improves the performance of the classifiers. These are the criteria:

$$\begin{array}{ll} \text{num_pixel_in_shower} \geq 10 & \text{num_islands} < 8 \\ \text{length} < 70 \text{ mm} & \text{width} < 35 \text{ mm} \\ \text{leakage1} < 0.6 & \text{leakage2} < 0.85 \end{array}$$

6.1.4 Comparison of Observations and Simulations

The Achilles heel of any analysis depending on simulated data is how well the simulation describes reality. In this section, observed and simulated image parameter distributions are compared. Optimizing the simulation configuration is a slow and computationally expensive task. Root causes of disagreements between observed and simulated data are rarely obvious and recomputing the simulations takes considerable time.

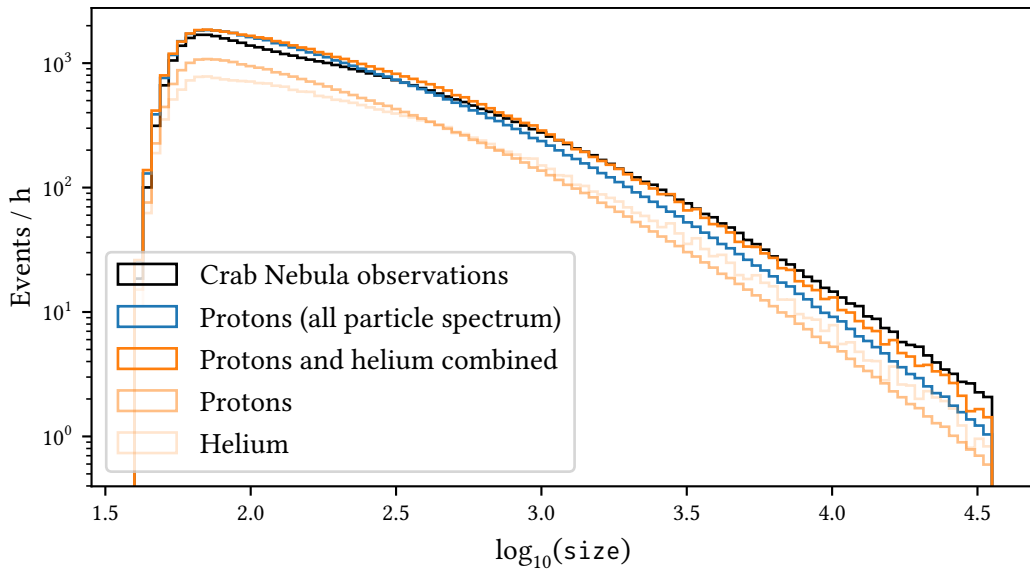


Figure 6.1: Image parameter size for the three different datasets normalized to the expected event rate for one hour of observation time. The simulated datasets with $APA = 85\%$ are used here.

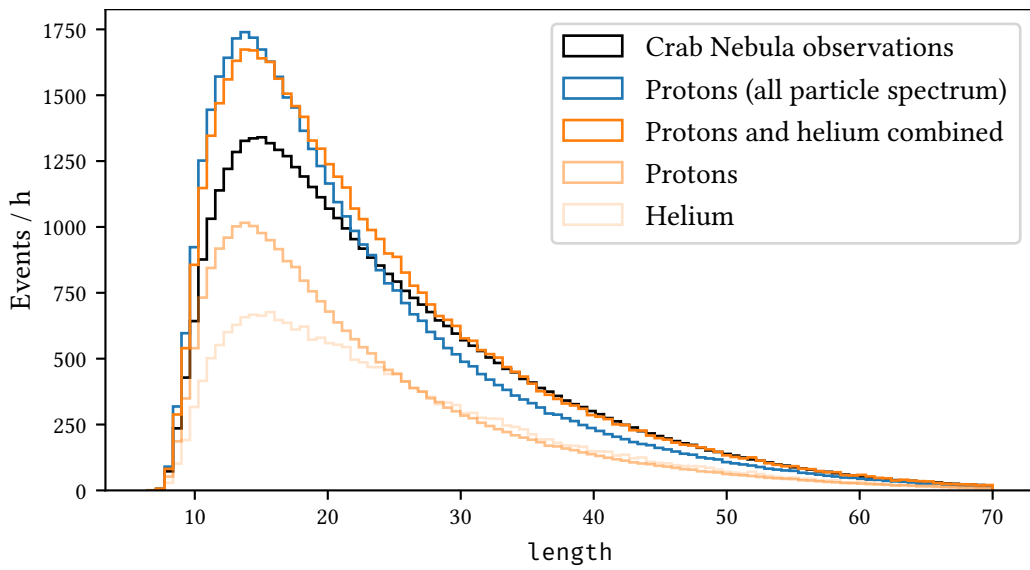


Figure 6.2: Image parameter length for the three different datasets normalized to the expected event rate for one hour of observation time.

Before, only protons had been simulated for FACT and been reweighted to (1.5), basically making the assumption that all charged cosmic rays are protons, e. g. in [149].

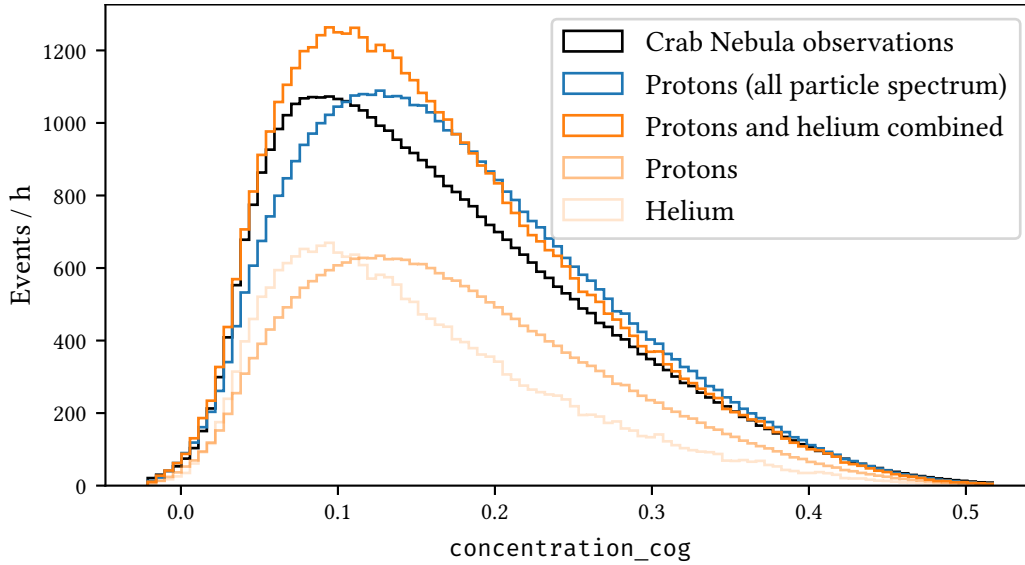


Figure 6.3: Image parameter `concentration_cog` for the three different datasets normalized to the expected event rate for one hour of observation time.

In Figure 6.1, the distributions of the size parameter, the total number of reconstructed Cherenkov photons in the cleaned image, is shown for three datasets:

1. The observed Crab Nebula observations
2. Simulated protons using $\text{APA} = 85\%$ weighted to the all particle spectrum (1.5).
3. Simulated protons and helium using $\text{APA} = 85\%$ weighted to their respective individual spectra according to (1.6) and (1.7) as shown in Figure 1.8.

As can be seen, the effect of the composition of cosmic rays is not negligible. For larger values of size the agreement between observations and the combined proton and helium flux is much better than under the only protons assumption. There are however considerably more events in both simulated distributions at smaller values of size compared to the observations. Mismatches are also observed in the other image parameters, e. g. `length` (Figure 6.2) or `concentration_cog` (Figure 6.3).

To check whether the mismatches in the other image parameters are correlated to the mismatches observed in size, Figure 6.4, Figure 6.5 and Figure 6.6 show the image parameters `length`, `concentration_cog` and `leakage1` only for events with $\text{size} \geq 500$. The agreement is much better for these bright events, hinting that the mismatches mostly affect the lower energy part. This can have multiple causes, dimmer events have a lower signal to noise ratio, resulting in not properly simulated noise affecting these events more strongly. Noise is simulated under the simple assumption of a single white noise component in `ceres`, compare section 3.2. An approach to mitigate the simple noise model in `ceres` is to use measured noise and superimpose it to simulated shower images without noise to get a more realistic noise for simulated events. A first implementation of this is described in [50] and is currently being evaluated for FACT simulations in [52].

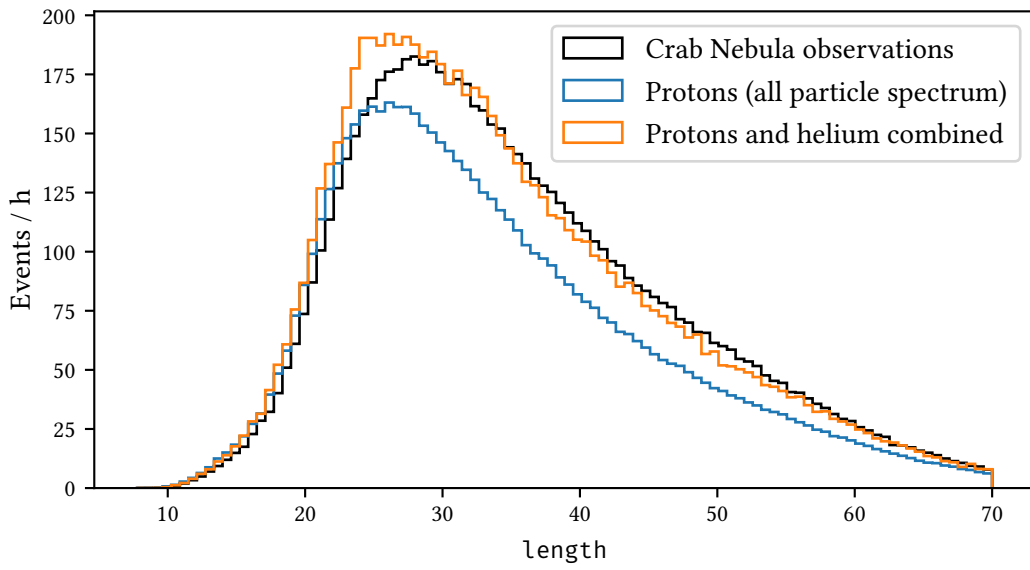


Figure 6.4: Image parameter length for the three different datasets normalized to the expected event rate for one hour of observation time. Only events with size ≥ 500 are shown.

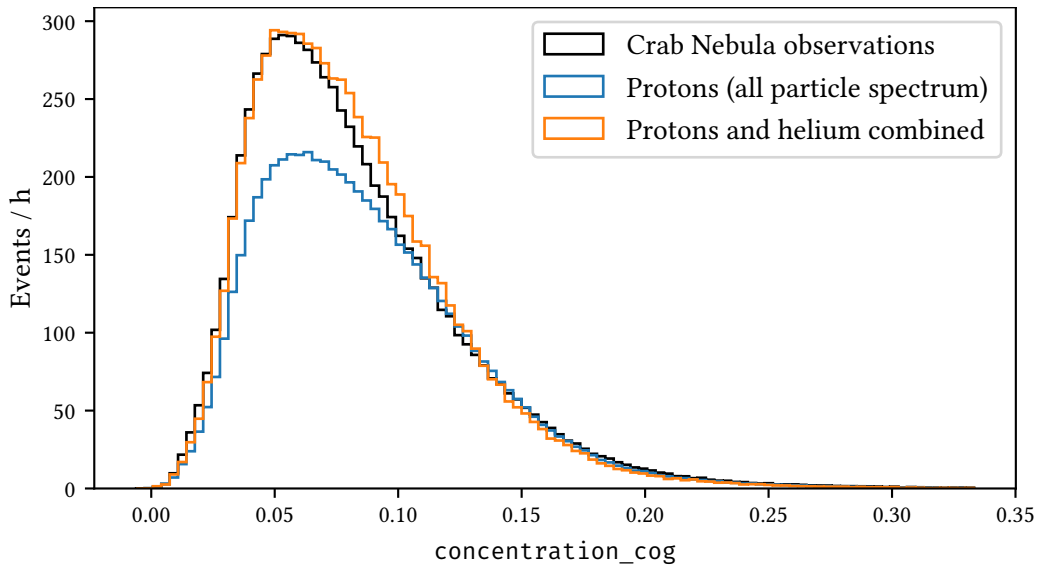


Figure 6.5: Image parameter concentration_cog for the three different datasets normalized to the expected event rate for one hour of observation time. Only events with size ≥ 500 are shown.

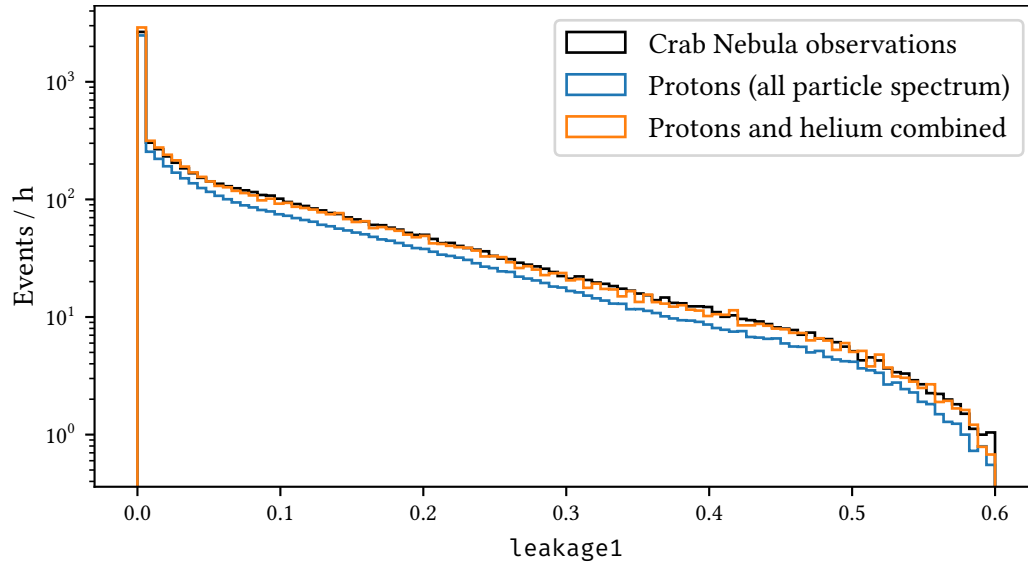


Figure 6.6: Image parameter `leakage1` for the three different datasets normalized to the expected event rate for one hour of observation time. Only events with size ≥ 500 are shown.

Figure [Figure 6.7](#) shows the image parameter size for all three simulated settings of the additional photon acceptance. As expected, with more light in the events, more events are triggering and overall higher event rates are observed, especially at lower values of size.

Finally, [Figure 6.8](#) shows a comparison between signal candidates on the observed Crab Nebula data and simulated gammas for the dataset with $\text{APA} = 85\%$. This is achieved by applying very loose event selection criteria ($\text{gamma_prediction} \geq 0.5$ and $\theta^2 \leq 0.1 \text{ deg}^2$) and subtracting the distribution for the off regions from the distribution in the on region, leaving only the excess events. The same distributions for the other settings of APA are shown in [Appendix 12](#). Also these figures lead to the conclusion that of the three simulated datasets, $\text{APA} = 85\%$ has the best agreement with the observed data, at least in the higher energy range.

These comparisons show, that work is still required, especially for the lower energy events, to get to full agreement between observed and measured distributions. Through the addition of helium overall agreement has been greatly improved, allowing more dedicated searches for the causes of the remaining disagreements in the future.

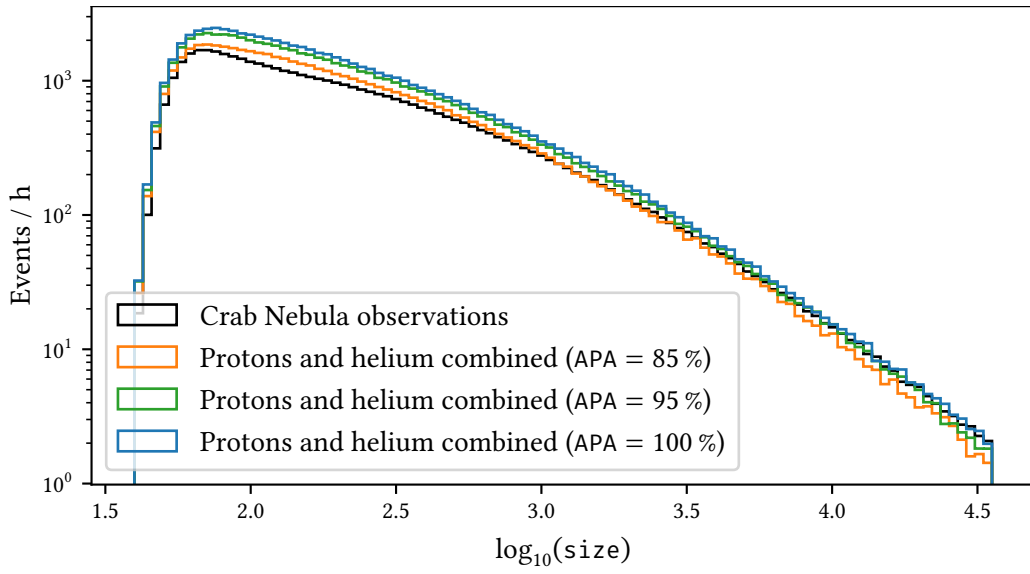


Figure 6.7: Image parameter size for the Crab Nebula observations and combined proton and helium flux for all three simulated values of APA.

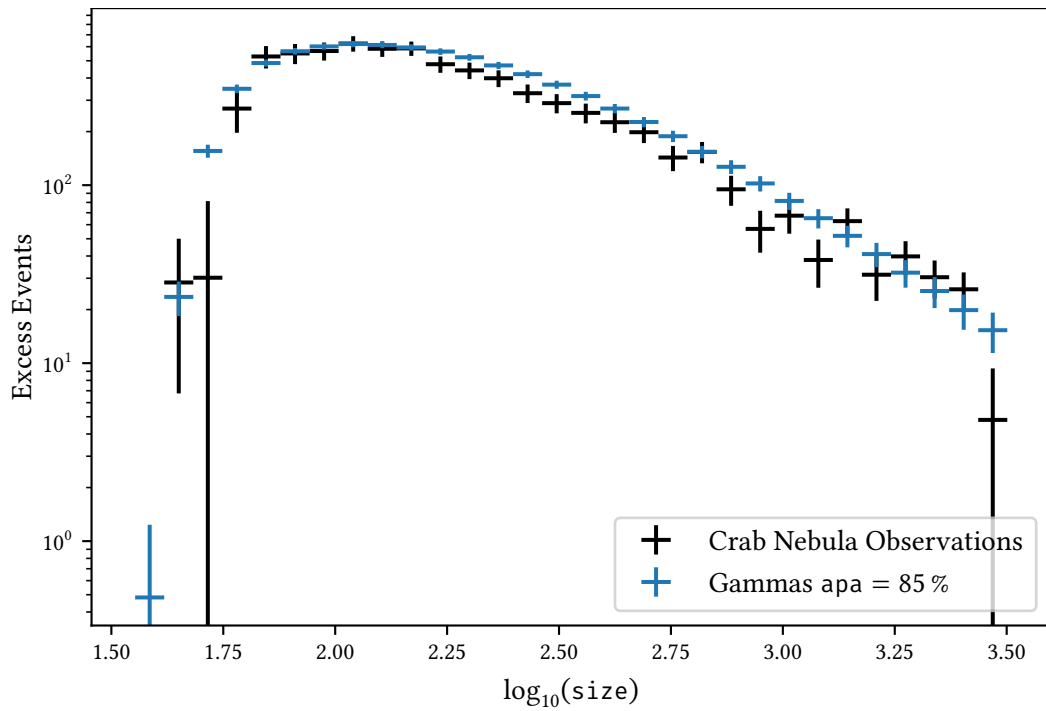


Figure 6.8: Image parameter size for simulated gammas and the observed excess events in the on region for loose event selection criteria. The simulated datasets with APA = 85% are used here.

6.2 Detection Significance of the Crab Nebula

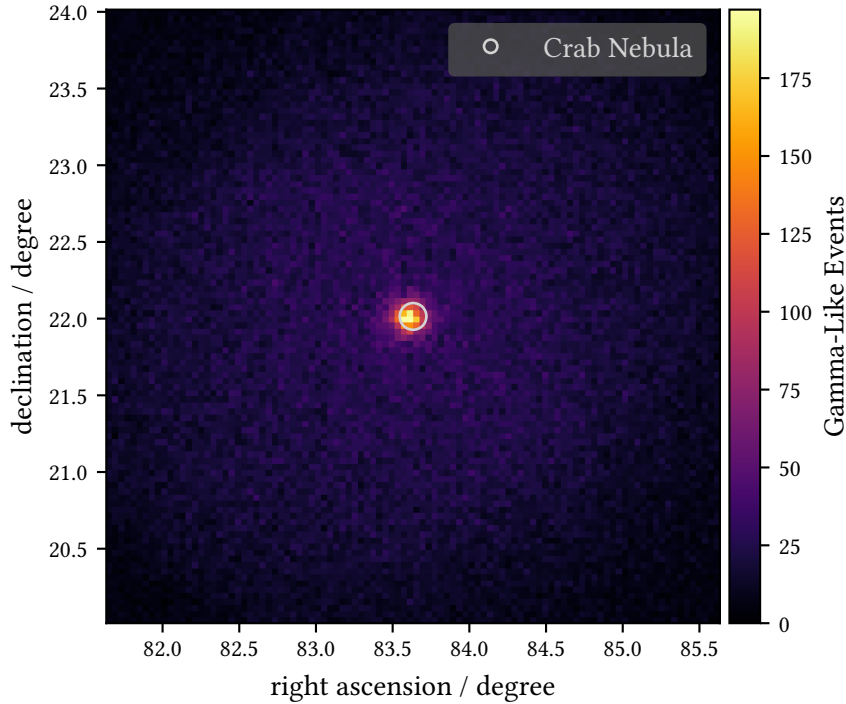


Figure 6.9: Skymap of the reconstructed positions of all events after applying a prediction threshold of $t_p = 0.82$. The catalog position of the Crab Nebula is marked using a gray circle. The reconstruction models were trained on the $APA = 85\%$ dataset. A small systematic offset of the center of gravity of the reconstructed events vs. the catalog position is visible.

After training the four models for the event reconstruction on the respective simulated datasets, they are now applied to the observed Crab Nebula data and the simulated datasets for the performance evaluation. The source dependent parameters like the distance of the reconstructed source position to the assumed source positions are also computed in this step. The result are lists of reconstructed events with very few properties per event, a last dramatic reduction of needed space:

- Event identifiers
- Time of observation
- Estimated energy
- Estimated source position in equatorial (ICRS) coordinates.
- Score of the gamma-hadron-separation
- Pointing direction of the telescope
- Angular distances θ to the observed source and the five off positions

FACT uses an event identifier composed of three keys: the night when observations were started (the date of the previous evening if after 00:00), the run number within that night and the event number within the run. Together these three form the unique identifier of an event recorded by FACT. Simulations only use the corsika run number and the event number.

6 Instrument Response Functions and Sensitivity of FACT

As described in section 2.2, FACT observes sources in wobble mode. This allows simultaneous estimation of the background rate at positions in the field of view geometrically identical to the source position. All events within an angular distance of θ_{\max} of the suspected source position are counted as “on” events and all events within that radius of any of the off positions are counted as “off” events, used for estimation of the remaining background.

The statistical significance of a source detection can be calculated using the likelihood ratio test first proposed by Li and Ma in [108] testing the null hypothesis of no source of gamma rays. The null hypothesis is rejected with a significance in units of standard deviations σ given by

$$S = \sqrt{2} \left(N_{\text{on}} \ln \left(\frac{1 + \alpha}{\alpha} \cdot \frac{N_{\text{on}}}{N_{\text{on}} + N_{\text{off}}} \right) + N_{\text{off}} \ln \left((1 + \alpha) \cdot \frac{N_{\text{off}}}{N_{\text{on}} + N_{\text{off}}} \right) \right)^{1/2} \quad [108, (17)] \quad (6.1)$$

where N_{on} is the number of events recorded in the on region, N_{off} is the number of events in all off regions and α is the size of the on region divided by the size of the Off region. The FACT-Tools analysis uses 5 identical Off regions, so $\alpha = 0.2$. By convention in particle physics, 3σ are interpreted as a strong hint and 5σ are called a detection, corresponding to false alarm probabilities of 1.3×10^{-3} and 3×10^{-7} , respectively. In the case for this analysis however, the goal is not detecting a new, unknown source but to estimate the potential of the telescope and the analysis.

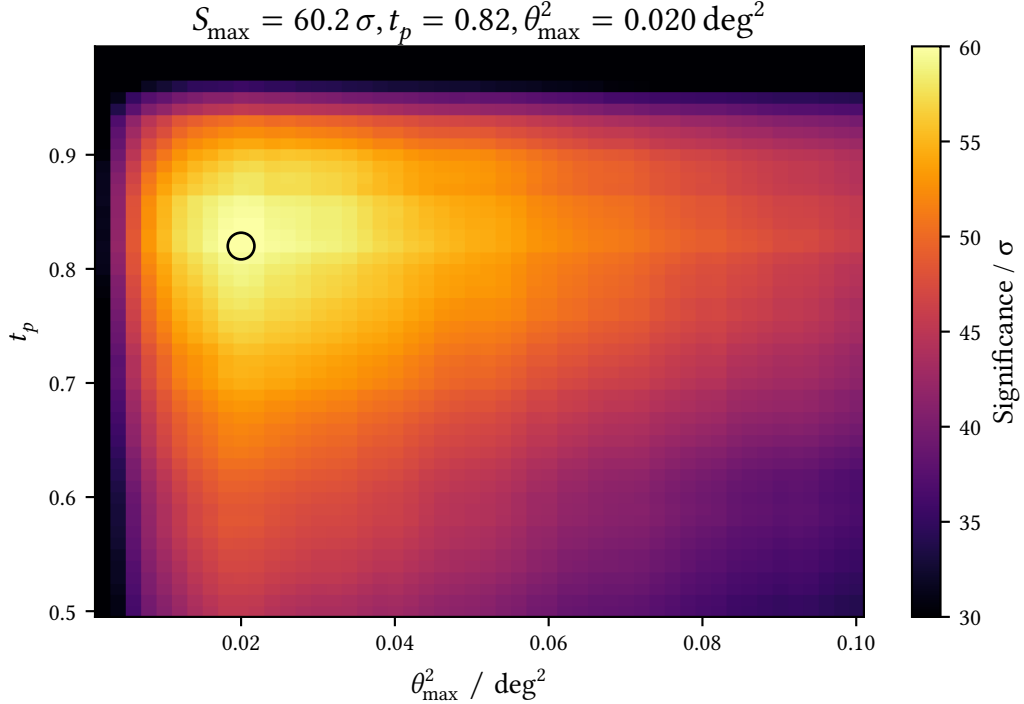


Figure 6.10: Significance for the detection of the Crab Nebula for a grid-search of event selection parameters using the simulated data with APA = 85 % for training the models. t_p was varied between 0.5 and 1.0 in steps of 0.01 and θ_{\max}^2 was varied between 0 and 0.1 in steps of 0.02. The best event selection in terms of detection significance according to (6.1) is marked using the black circle.

The final step is applying the prediction threshold t_p for the gamma-hadron-separation and

choosing the radius of the on and off regions θ_{\max} . This is done using a grid-search, shown in Figure 6.10, calculating the significance for each combination of t_p and θ_{\max} .

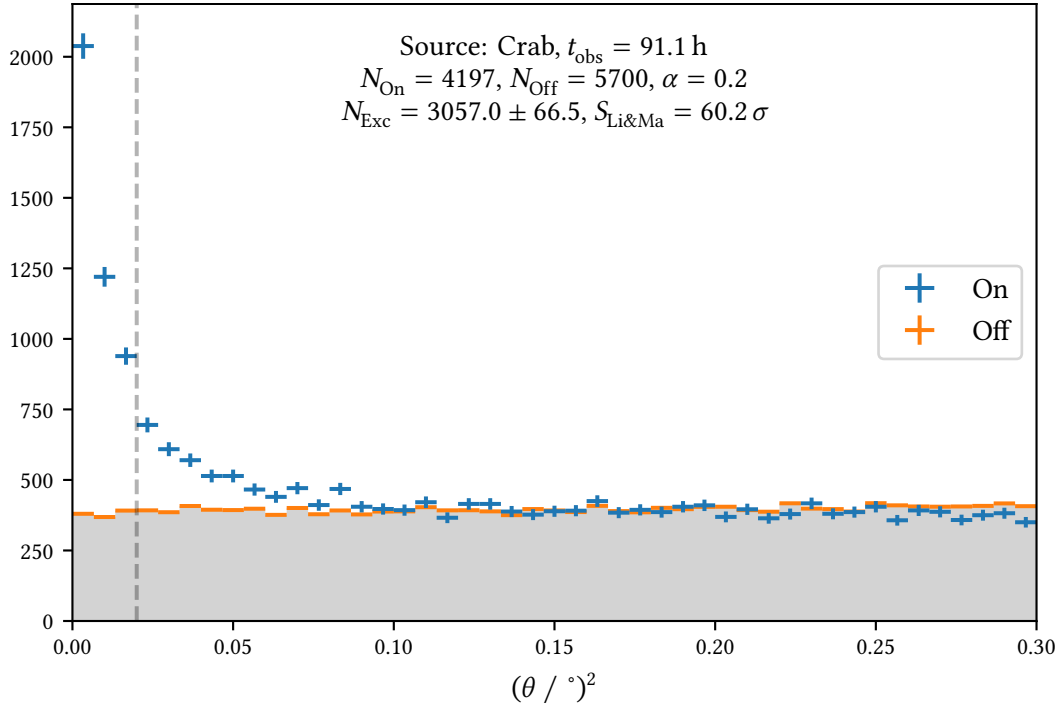


Figure 6.11: Theta-Squared-Plot for the Crab Nebula dataset reconstructed using the models trained on the simulated dataset using $\text{APA} = 85\%$ and using the t_p and θ_{\max} yielding the highest significance from Figure 6.10.

With a maximum significance of over 60σ , this is a major improvement over the previous analyses. On the same dataset, the RapidMiner based analysis achieved 39.9σ [149, page 72] and an analysis using the `aict-tools`—before the new `disp` estimation was implemented—achieved 43.3σ [128]. The onsite quick-look-analysis based on `MARS` achieved 41.3σ ¹.

In gamma-ray astronomy, the most iconic visualisation for the detection of a source is the θ^2 -Plot. It shows the distribution of the angular distances to the assumed source position for both the source of interest and for all off positions combined. In case a detection is made, an excess of events in the on region above the off region is expected for very small values of θ^2 . This can be observed beautifully in Figure 6.11.

In Table 6.2, the number of events after each analysis step is shown for all datasets.

¹Results taken from FACT’s database.

Table 6.2: Event numbers after each step of the analysis for the five different datasets. Numbers after applying the threshold and the on region selection are those for the proton and gamma simulations are those of the test datasets, i. e. only for 75 % of the events. No test dataset is created for the diffuse gamma rays.

Dataset	Triggered	Image Cleaning	Pre-Selection	Separation	On-Region
Crab Observations	21898338	7953392	4055394	236444	4197
Diffuse gammas	1491524	1178551	912643	—	—
Wobble gammas	11304586	8632556	6449807	1590751	1122345
Protons	8709718	6701859	5307824	100578	457
Helium	314812	243472	195375	759	4

6.3 Separation Performance

The random forest classifiers to separate gamma ray induced showers from hadron induced background were trained using the `aict-tools` configuration in Appendix 11. It uses 500 000 samples each of the signal and background class and 18 features in total. A useful property of tree based models is their ability to provide information about which features contributed the most to the decision making. Figure 6.12 shows the feature importances for each of the 200 decision trees in each of the 10 cross validation steps.

The ROC-curves for each of the cross validation iterations and the mean are shown in Figure 6.13. Figure 6.14 shows the area under the ROC-curve in intervals of estimated gamma energy as predicted by the energy prediction model. It can be seen that the larger the estimated energy, the better the model can distinguish between gammas and protons in the same energy interval.

In the next section, the effective collection area for gammas and protons is shown in Figure 6.19, demonstrating that applying the selection criteria dramatically decreases collection area for protons while keeping most of the gamma rays.

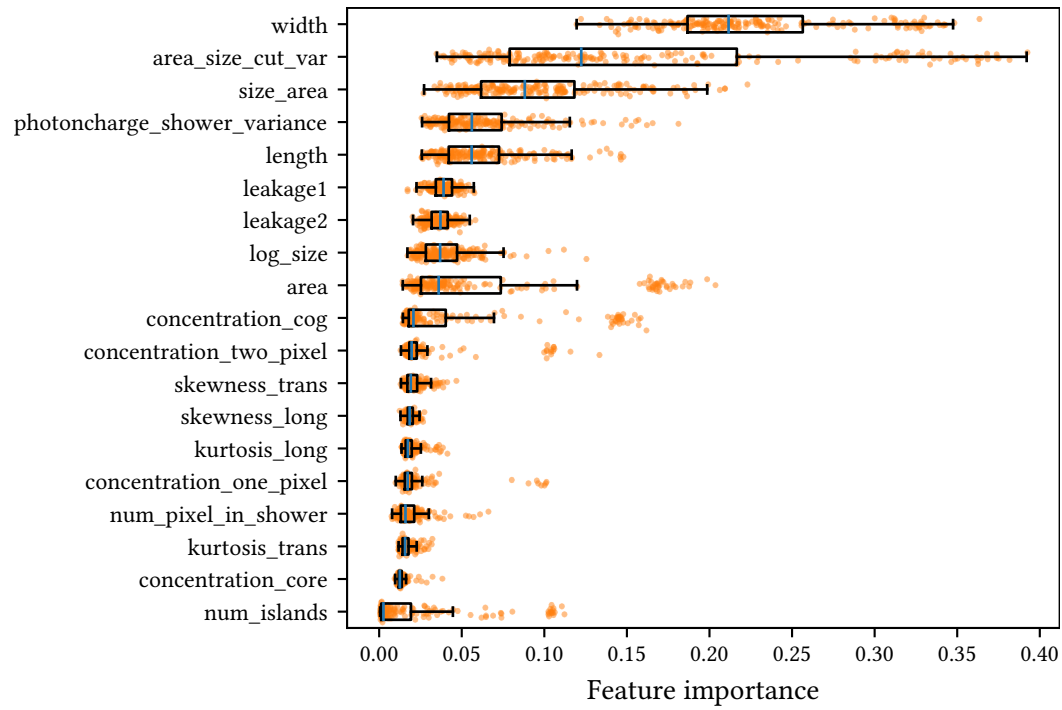


Figure 6.12: Feature importances for the classification model. Each point represents one tree in one of the cross validation iterations, the box plots show the median as blue line, the size of the box goes from the first to the third quartile and the whiskers extend. The feature `area_size_cut_var` has been used for manual event selection in the MARS analysis and is defined as $\text{area}/(\log \text{size})^2$, where $\text{area} = \pi \cdot \text{width} \cdot \text{length}$. `size_area` is defined as size divided by area. Both of these features are generated automatically by the `aict-tools` from the input features described in [section 4.6](#).

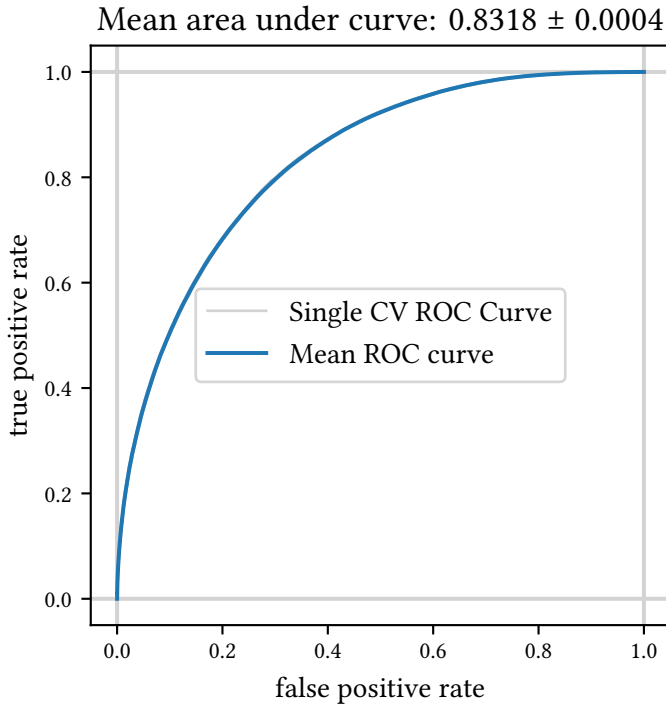


Figure 6.13: ROC curves of the classification model. Because the number of training samples is very large, variance between cross validation iterations is low and the single curves are only barely visible.

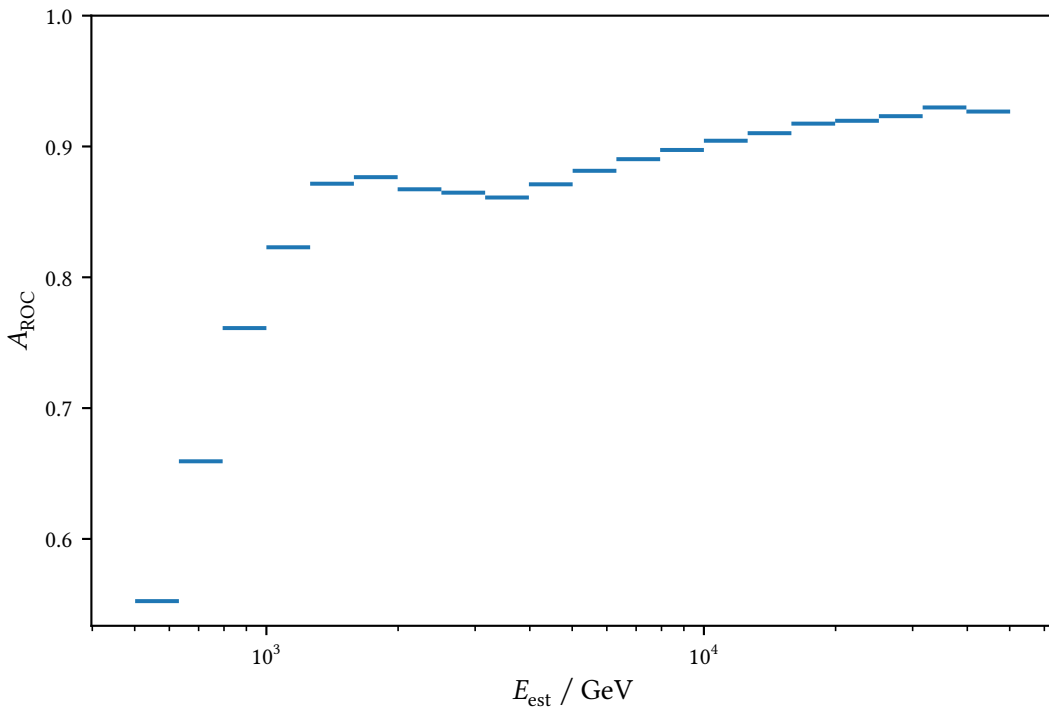


Figure 6.14: Area under the ROC-curve evaluated in bins of estimated gamma energy E_{est} . The higher the estimated energy, the better the classifier performance. For a discussion, why bins of estimated energy are used see [section 6.5](#).

6.4 Instrument Response Functions

Instrument response functions (IRFs) are key properties of the combined system of detector and analysis software that describe the response to particles given their physical attributes. The IRFs are needed to do any physics analysis from lists of events with reconstructed properties like estimated energy, particle type and origin with additional metadata such as measurement time and observed source. Four different IRF components together fully describe the measurement process of any IACT:

- Effective area, [subsection 6.4.2](#), describes the detection efficiency of a telescope as the area a perfect detector directly measuring gamma rays with the same efficiency as the real telescope would have.
- Energy dispersion, [subsection 6.4.3](#), describes the performance of the energy reconstruction as the probability density to get an estimated energy E_{est} for a given true energy E_{true} .
- Point spread function (PSF) of the gamma ray origin reconstruction, [subsection 6.4.1](#), not to be confused with the optical point spread function of the reflector system, describes how well the position of a gamma ray can be reconstructed.
- Background rate describes the number of background events still present after applying the event selection after the gamma-hadron-separation.

For a long time, exact definition and treatment of these were slightly different from experiment to experiment but with the advent of CTA and it being operated as an open observatory, a standardization effort has started under the name “Data Formats for Gamma-Ray Astronomy” [58]. In their most general form, called full-enclosure-IRFs by [58], all these depend on the energy, either true or estimated, and the offset of the reconstructed source position from the optical axis, relying on the assumption that the IRFs are radially symmetric.

In case of observing point sources in wobble mode, the IRFs are simpler, as only one offset from the optical axis needs to be given, the background can be estimated from the same observations and the effective area includes the selection of the on region. The PSF only plays a role in the selection of the on region’s size and is not needed afterwards. Only these point-like IRFs will be discussed here, as they are the most relevant for FACT’s monitoring of well-known, bright point sources.

The calculation of these IRFs was implemented in a python package of the same name, again by Kai Brügge and myself². The uncertainties are estimated by applying the same calculations to datasets created by sampling with replacement from the original dataset, bootstrapping.

6.4.1 Angular resolution

The angular resolution is a measure for how well an analysis can reconstruct the origin of a gamma ray. As the `disp` parameterization is now done using the `aict-tools`, the used features and the parameters for the random forest are stored in the configuration file in [Appendix 11](#). [Figure 6.15](#) shows the performance metrics of the `disp` estimation in several energy ranges for simulated gamma-ray events before and after applying the prediction threshold. Both accuracy and r^2 score

²<https://github.com/fact-project/irf>

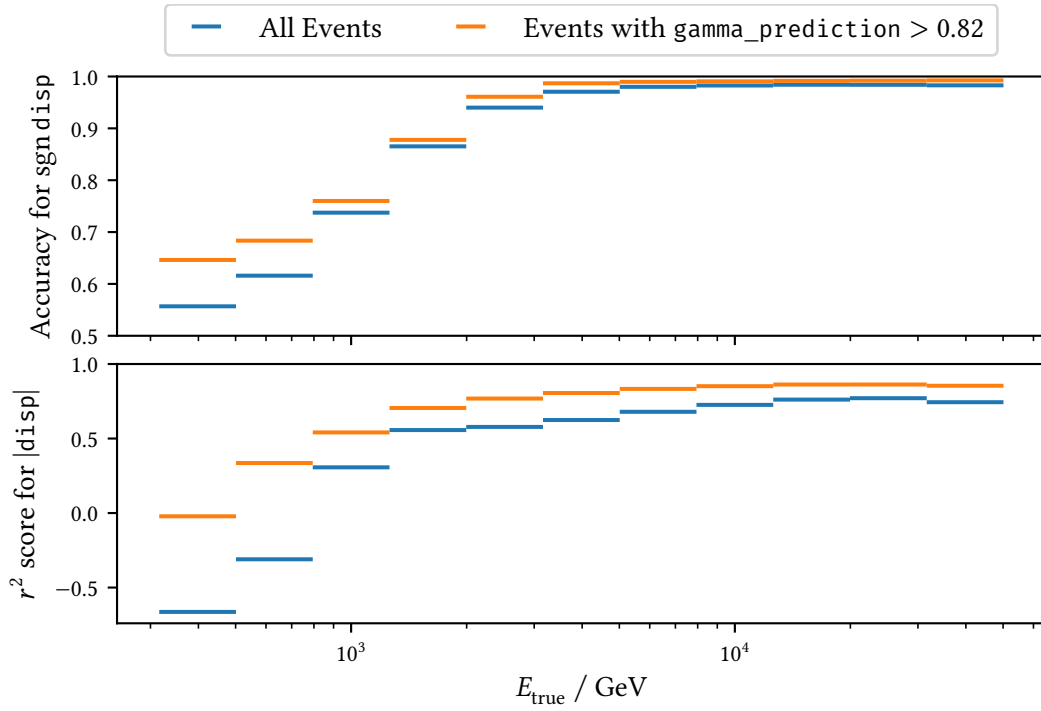


Figure 6.15: Accuracy of the prediction for sgn disp and r^2 score of the prediction for |disp| calculated in bins of true energy for the point source gammas observed in wobble mode before and after applying the prediction threshold.

improve with higher energies and gamma-ray events that earn a high `gamma_prediction` are also easier to reconstruct for the `disp` prediction.

Compared to the older FACT-Tools parameterization, the angular resolution has drastically improved, especially in the higher energies. While at the lower energies more events are lost due to misclassification of `sgn disp`, the background rate was reduced by a factor of two as the diffuse background is not skewed towards the assumed source position anymore.

Angular resolution is defined here as the radius from the source position which contains 68 % of the reconstructed events.

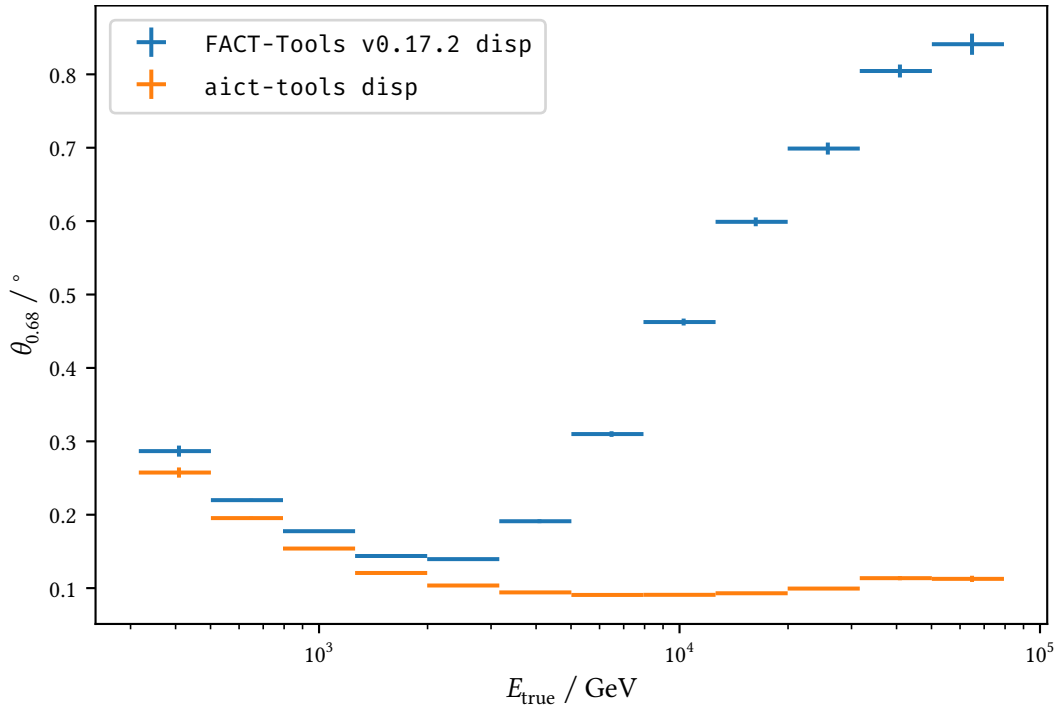


Figure 6.16: Comparison of the angular resolution using the simple `disp` parameterization from FACT-Tools with the new, machine learning based parameterization implemented in the `aict-tools`. Large improvements could be achieved over the full energy range, but especially at the higher energies. Only events with the correct prediction for `sgn disp` and with `gamma_prediction` ≥ 0.020 were used. A version where all events were used is shown in Appendix 13, showing that for small energies, a lot of events are essentially lost because of the wrong sign prediction. This can also be seen in Figure 6.15 showing the accuracy of the sign prediction and in Figure 6.17 when comparing the effective area before and after applying the event selection based on θ .

Figure 6.16 shows the angular resolution of the dataset with $\text{APA} = 85\%$ for the old `disp` parameterization (5.10) compared to the new random forest based approach. The random forest based approach improves the angular resolution over the full energy range, but the improvement is particularly large for energies higher than 5 TeV.

6.4.2 Effective Area

The differential (in energy) flux of a gamma-ray point source has the dimensionality particles per time, area and energy. As a telescope counts gamma rays, the observed area, the observation time and the energy of the particles needs to be known to estimate flux. As a real telescope is also unable to detect all gamma rays that reach the telescope, it is necessary to know the detection efficiency or acceptance to correctly reconstruct the original number of gamma rays. In gamma ray astronomy, it is common to define an effective area A_{eff} that combines the area the detector is observing with the detection efficiency. It can be interpreted as the area a perfect detector directly observing the gamma rays would need to observe the same number of particles. Effective area can only be expressed as function of the true, simulated energy, as it relates the number of detected particles to the number of all particles and no estimated energy exists for the non-detected particles.

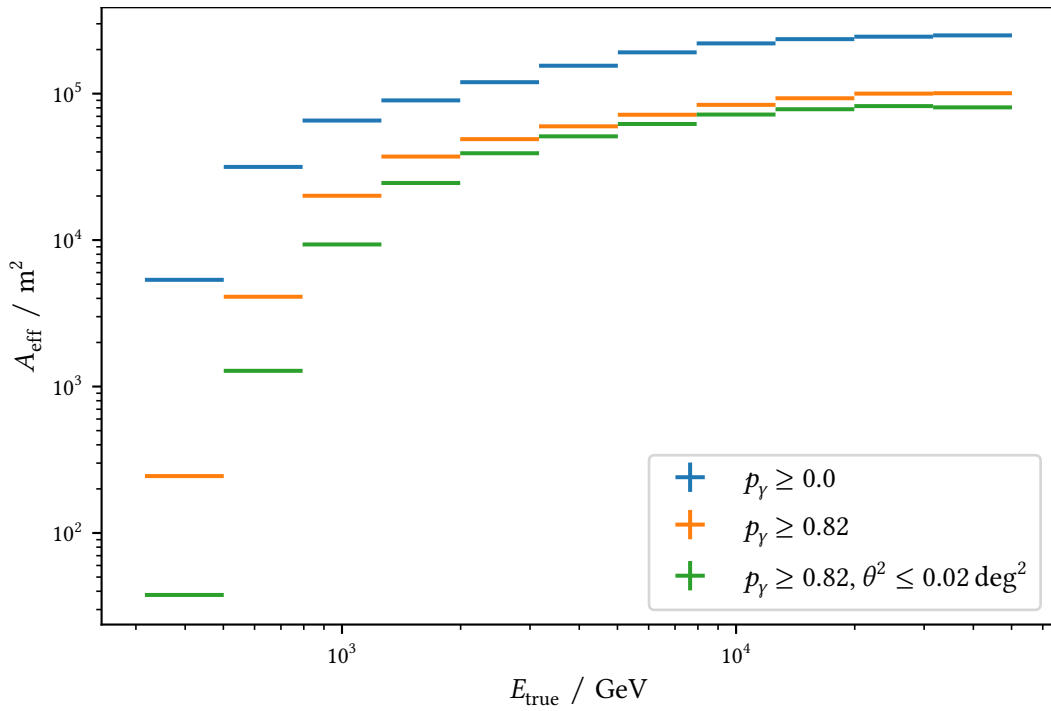


Figure 6.17: Effective area in bins of simulated energy for gamma rays from a point source observed in wobble mode after the event pre-selection, after applying the prediction threshold and after applying both the prediction threshold and the selection using θ_{max}^2 . The efficiency of the classifier results in an overall drop in effective area while the θ^2 cut mainly reduces effective area for low energies.

Effective area is then defined as

$$A_{\text{eff}}(E) = p(E)A \quad (6.2)$$

where $p(E)$ is the detection probability for a gamma ray with energy E and A is the area the telescope is able to observe. $p(E)$ is estimated by counting detected events N_{detected} and divide by the total number of simulated events $N_{\text{simulated}}$ in intervals of E . Thus the discretized effective area in the i -th energy interval is:

$$A_{\text{eff},i} = \frac{N_{\text{detected},i}}{N_{\text{simulated},i}} A_{\text{simulated}} \quad (6.3)$$

$A_{\text{simulated}} = \pi R_{\text{max}}^2$ is the area the impact points of the gamma rays were simulated in. The simulation has to sample the whole area, the telescope is sensitive to, as otherwise the effective area will be underestimated. Figure 6.18 shows the distribution of impact distance of the simulated gamma rays after applying the image cleaning.

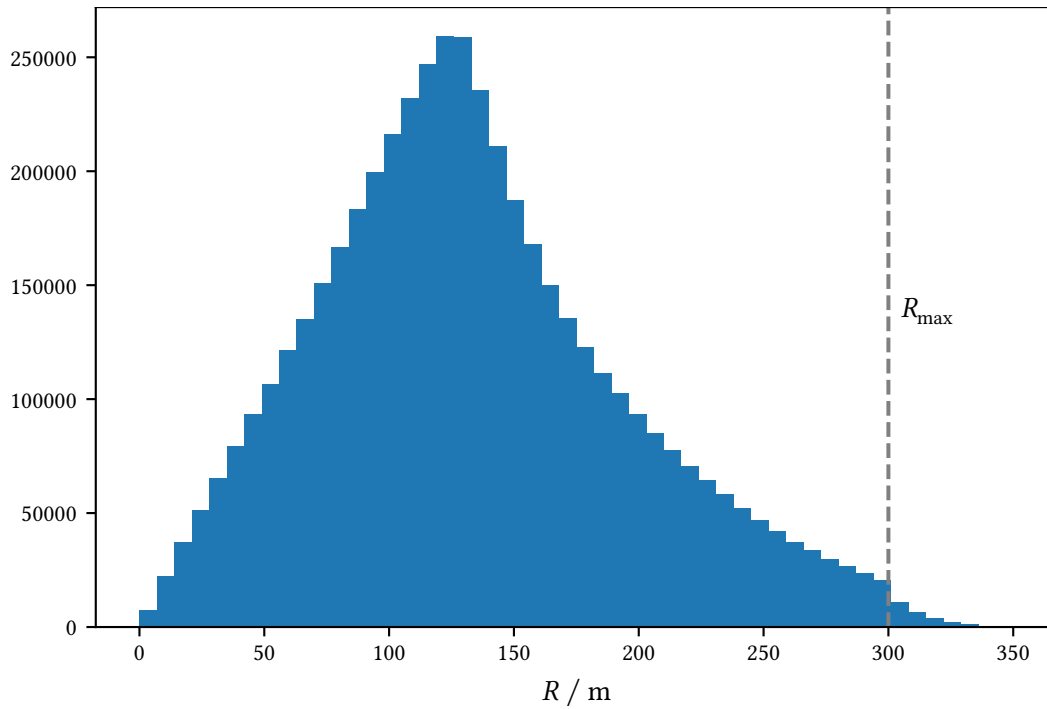


Figure 6.18: Distribution of the impact parameter, the distance from where the shower axis intersects the observation level to the telescope. Because CORSIKA draws the impact scattering perpendicular to the shower axis but measures the impact point on the ground level, the cutoff is not hard at $R_{\text{max}} = 300$ m but smeared out a little due to events with zenith distances > 0 . The number of detected events rises linearly first, with the increasing area enclosed by the respective ring, but drops of after roughly 130 m, which is related closely to the radius in which the main part of the Cherenkov light reaches the observation level, compare Figure 1.2.

The effective area can be evaluated for different steps in the analysis, where each step is additionally discarding events:

1. Events that triggered the telescope, see section 2.1.
2. Events with more than five pixels after image cleaning, see section 4.5.

6 Instrument Response Functions and Sensitivity of FACT

3. Events surviving the event pre-selection, see [subsection 6.1.3](#).
4. Events after applying the gamma-hadron-separation threshold $p_\gamma > t_p$, see [section 5.2](#).
5. Events in the On region after applying $\theta^2 < \theta_{\max}^2$, see [section 6.2](#).

Effective area for the last three steps is shown in [Figure 6.17](#) for gamma rays from a point source observed in wobble mode. [Figure 6.19](#) shows the effective area for step 3 and 5 also for the background class. The shape is as expected in gamma-ray astronomy, first rising steeply with energy and then reaching a plateau with a slight drop off at the highest energies caused by the growing percentage of showers not fully contained in the camera.

It is only possible to evaluate effective area with respect to the true gamma-ray energy as no estimated energy is known for events that did not trigger the telescope, survive the image cleaning or were discarded by the event pre-selection.

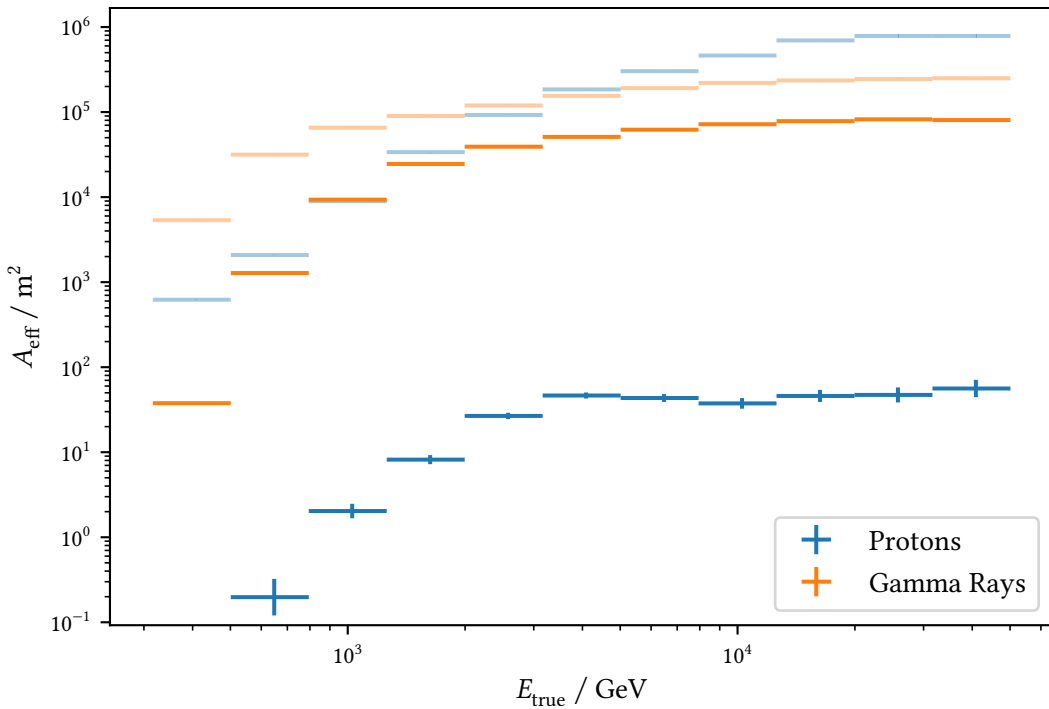


Figure 6.19: Effective area in bins of simulated energy for gamma rays from a point source observed in wobble mode and protons after the event pre-selection (transparent) and after applying both the prediction threshold and the selection using θ_{\max}^2 (solid). The first energy bin did not contain any proton events after applying the event selection criteria and is thus not shown in the logarithmic scale. While overall roughly 25 % of the gamma rays are kept, the proton background is reduced by at least three orders of magnitude.

6.4.3 Energy Dispersion

Energy dispersion or migration is the probability density function of measuring an estimated energy E_{est} given a true energy E_{true} . Knowledge of the energy dispersion is necessary as the energy estimation is not perfect and in general has non-zero bias and resolution. The energy dispersion is thus needed for the statistical reconstruction of spectra from lists of reconstructed events. Methods for this will be discussed in [chapter 7](#). As the energy dispersion closely relates to the other performance metrics of the energy estimation, these will also be discussed here.

The logarithm of the energy is estimated using a random forest regressor with the configuration in [Appendix 11](#). It is trained on 500 000 diffuse gamma rays and comprises 200 single decision trees. The depth of the trees is limited and a minimum leaf size is set to prevent overtraining and overly large models, also achieving faster application times.

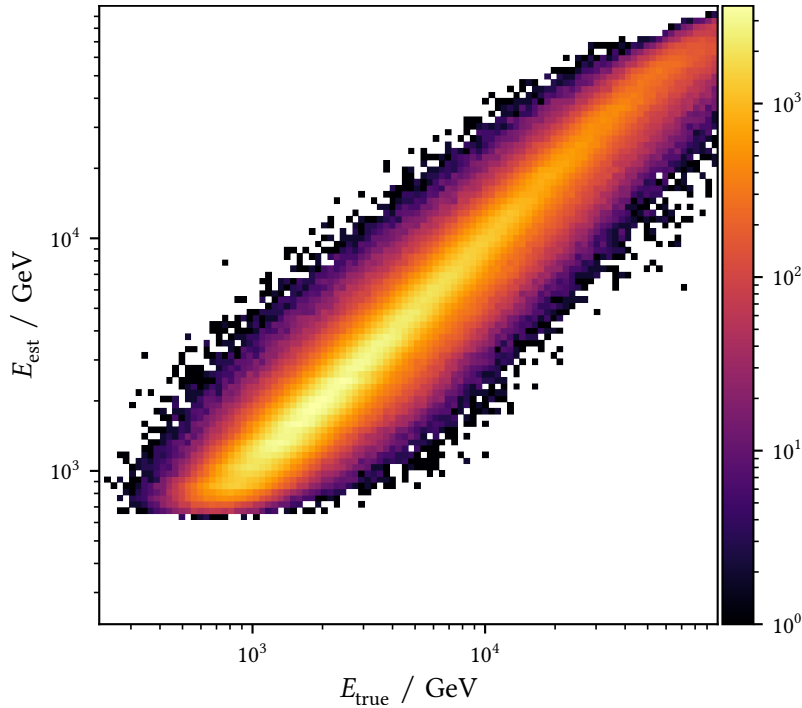


Figure 6.20: Energy migration matrix for gamma rays from a point source observed in wobble mode after applying the event selection.

On the cross validated training dataset, the regressor achieved an r^2 -score of 0.823 ± 0.006 . The migration matrix is shown in [Figure 6.20](#). As the r^2 -score is a single number over the full energy range used for training and dependent on the simulated spectrum in case the performance is not uniform over the energy range, two metrics are calculated in bins of true energy, both aggregating the relative error

$$\Delta E_{\text{rel}} = \frac{E_{\text{est}} - E_{\text{true}}}{E_{\text{true}}}. \quad (6.4)$$

Bias is here defined as the median of the relative error and *resolution* as half the interquantile distance between the upper and lower quantile containing 68.2 %, as the 1σ interval of the normal

6 Instrument Response Functions and Sensitivity of FACT

distribution, of the events:

$$\text{Resolution} = \frac{Q_{84.1}(\Delta E_{\text{rel}}) - Q_{15.9}(\Delta E_{\text{rel}})}{2} \quad (6.5)$$

Several other definitions³ exist and are used in publications, [21] is fitting a normal distribution to the histogram of the relative error. This however underestimates the resolution in case of skewed distributions, which occur regularly. CTA uses the interval centered around 0 containing 68.2 % of

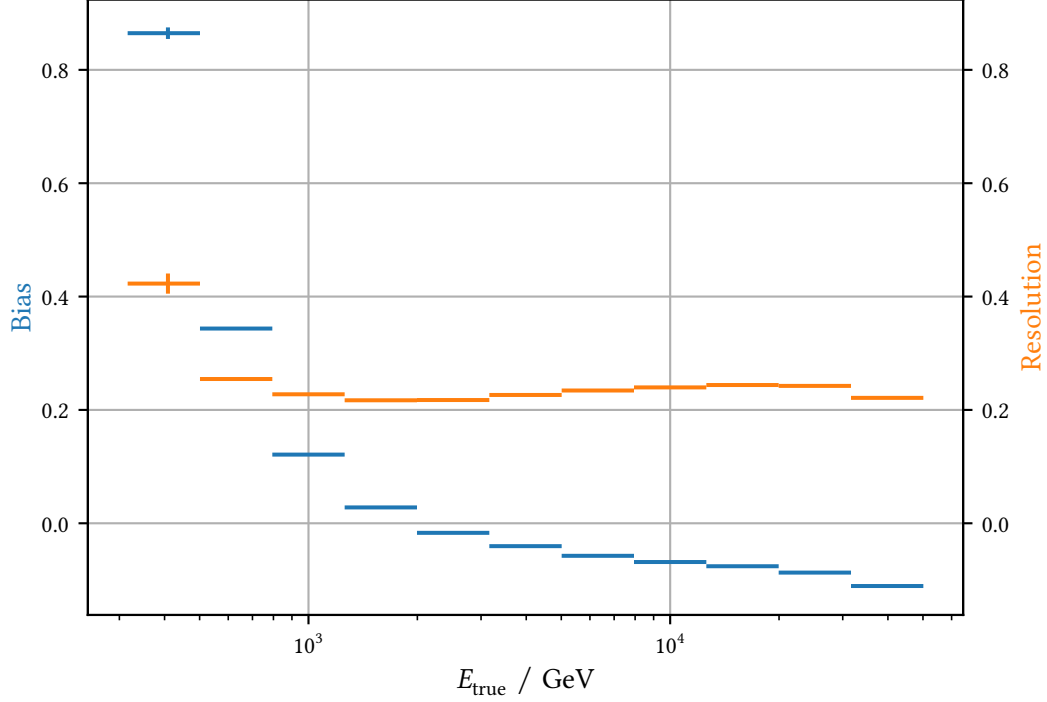


Figure 6.21: Bias and resolution for gamma rays from a point source observed in wobble mode after applying the event selection.

the events, which intertwines centrality and width of the relative error and can be interpreted as a quantile based version of the root mean squared. Just taking the mean and standard deviation is prone to be influenced heavily by few very large outliers, so the median and interquartile distance is taken as a more robust method of estimating bias and resolution. As with the angular resolution, the performance of the energy estimation improves quite dramatically after discarding gamma rays with low values for p_Y and only taking events in the On region into account. Bias and resolution for the events selected into the final dataset are shown in Figure 6.21, Appendix 14 shows bias and resolution without applying the event selection.

6.5 Differential Flux Sensitivity

The central measure of the performance of an IACT is its differential flux sensitivity, which is defined as the smallest flux the telescope can still detect with a certain statistical significance in a

³See <https://xkcd.com/927>

given observation time. Most commonly, IACT sensitivities are given for 5σ according to (6.1) in 50 h of observation time. As calculating the sensitivity requires counting events, the differential sensitivity is given as integral sensitivity in bins of energy.

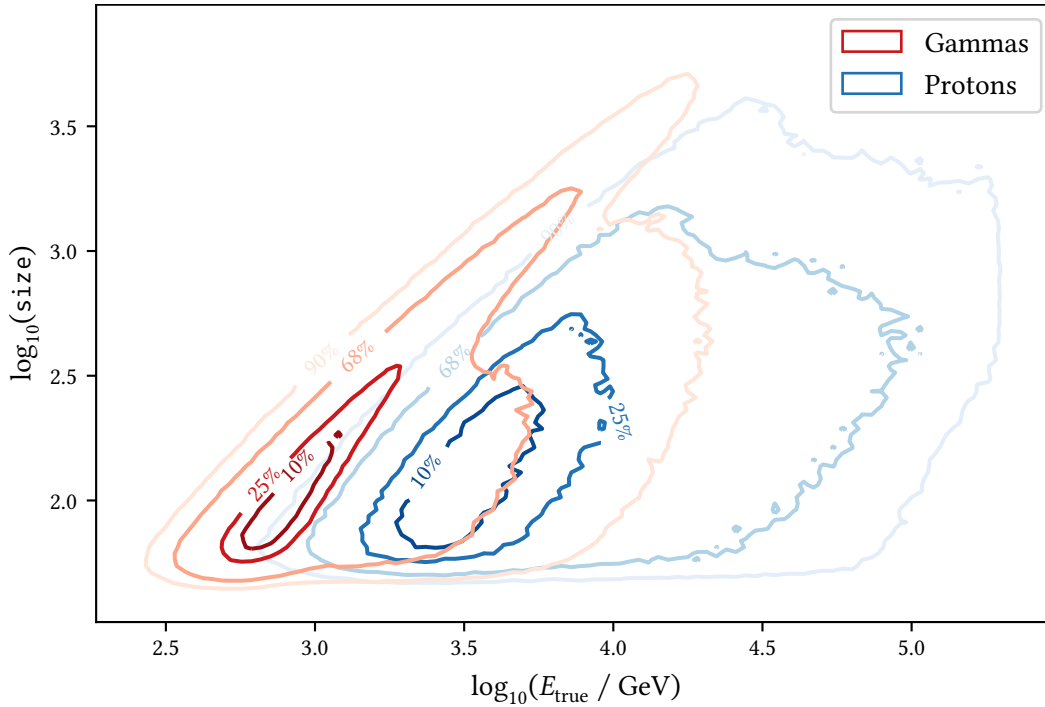


Figure 6.22: size vs. primary energy for simulated proton and gamma ray induced showers. The contours contain the listed percentage of all events. It can be seen, that protons need to have an approximately five times higher energy to produce the same amount of Cherenkov light in the camera of an IACT.

A controversial topic is whether the sensitivity should be calculated in bins of E_{true} or bins of E_{est} . Figure 6.22 shows the amount of Cherenkov light the FACT camera received, image parameter size, against the energy of the primary particle for gamma rays and protons. It is clear, that protons of the same primary energy as gamma rays do not produce similar images. The same can be seen in Appendix 16 which shows the same for the estimated gamma energy vs true energy. Consequently, calculating sensitivity in bins of true energy for both gammas and protons uses a very unrealistic background model and is also not possible to calculate on observed data. On the other hand, calculating sensitivity in bins of estimated energy is possible on both simulated and observed data and provides a more realistic background modeling in each bin. The disadvantage is that the behaviour of the energy migration plays a role in the discretization and that energy migration has to be taken into account to answer the question if a source with a certain flux would be detectable. A solution to both problems could be either to unfold the event counts using the energy migration, assuming unfolding the hadronic background yields a background estimation in bins of true energy or to only discretize the gamma rays in bins of true energy and find a better model for the background in these bins.

6 Instrument Response Functions and Sensitivity of FACT

For now, we look into the simulated and observed sensitivity in bins of estimated energies. In each bin, the number of events in the on region N_{on} and the number of events in the off regions N_{off} is calculated. The number of signal events expected from a source with flux of a fraction p of the Crab Nebula flux is estimated to be

$$N_{\text{signal}} = \frac{t_{\text{ref}}}{t_{\text{obs}}} (p \cdot (N_{\text{on}} - \alpha N_{\text{off}})), \quad (6.6)$$

where t_{ref} is the reference observation time for the sensitivity, t_{obs} is the observation time of the dataset and $\alpha = 0.2$ is the size ratio of the on region to the Off regions. The background expectancy only needs to be scaled to the correct reference time:

$$N_{\text{background}} = \frac{t_{\text{ref}}}{t_{\text{obs}}} N_{\text{off}} \quad (6.7)$$

The scaled event counts in the off and on region are then

$$N'_{\text{on}} = \frac{t_{\text{ref}}}{t_{\text{obs}}} (p \cdot (N_{\text{on}} - \alpha N_{\text{off}}) + \alpha N_{\text{off}}) \quad (6.8)$$

$$N'_{\text{off}} = \frac{t_{\text{ref}}}{t_{\text{obs}}} N_{\text{off}} \quad (6.9)$$

The flux sensitivity is then the value of p that solves

$$S_{\text{Li \& Ma}}(N'_{\text{on}}, N'_{\text{off}}, \alpha) \stackrel{!}{=} 5. \quad (6.10)$$

Unfortunately, finding p analytically is impossible and is thus done numerically using the Newton–Raphson method, the results are shown in [Figure 6.23](#).

For the simulated datasets, the gamma rays are again weighted to (1.4). Due to the limited helium core statistics, see [Table 6.2](#), the background is only modeled by protons weighted to (1.5). The sensitivity is calculated in five bins equidistant in log-space per decade in estimated energy for 50 h of observation time.

The integral sensitivity over the whole energy range is 8.5 % of the Crab Nebula flux for the observed data over the whole energy range, which is a considerable improvement over the 15.2 %⁴ in [149] and 13.7 % from [128].

⁴Calculated using (6.10) from the N_{on} and N_{off} given in [Table 6.7](#), p. 72

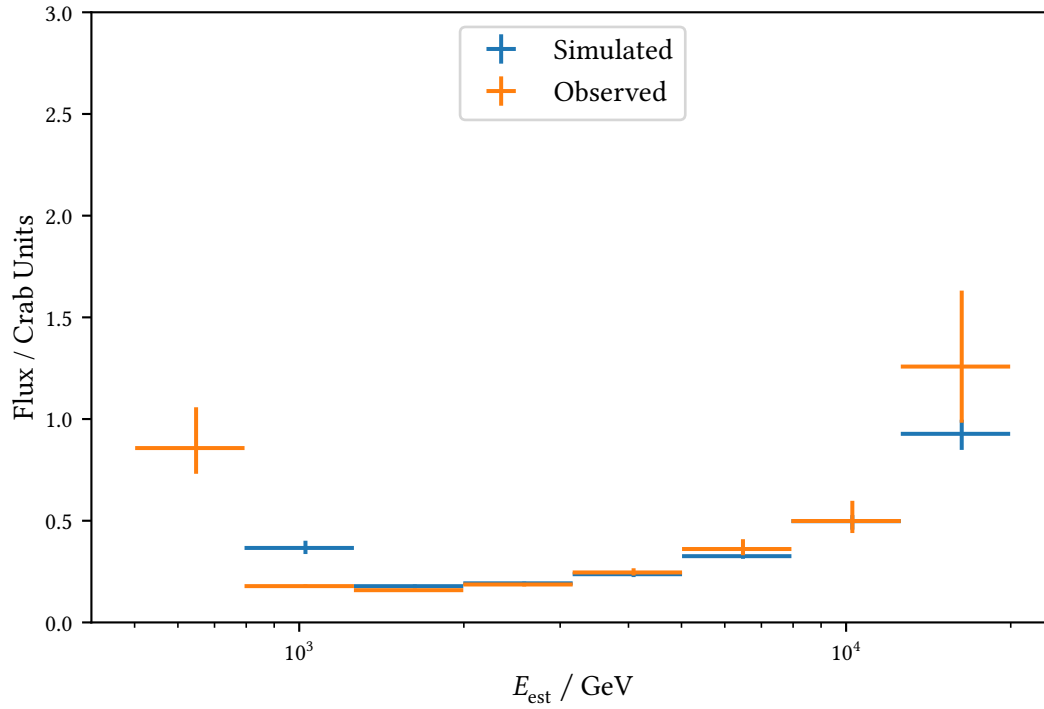


Figure 6.23: Differential sensitivity in bins of estimated energy E_{est} for simulated and observed data. For the simulations, the gamma-ray events were weighted to the flux of the Crab Nebula as measured and parameterized using a log-parabola by the MAGIC telescopes in [22]. The simulated protons were weighted to the all particle cosmic rays spectrum (1.5). While the simulated spectral sensitivities agree within statistical uncertainties for energies larger than 2 TeV, a disagreement is visible for the lower energies. The lowest energy bin is missing for the simulated dataset, as too few events remained here for estimating the sensitivity.

Energy Spectrum of the Crab Nebula

In the [previous chapter](#), the different properties of the FACT detector combined with the analysis presented in [chapter 4](#) and [chapter 5](#) were introduced. The data is now reduced to lists of events with estimations of their physical properties and some metadata like observation time and pointing direction of the telescope, both for the observed and simulated datasets. From these, the spectral energy distribution of the Crab Nebula will now be estimated.

This is the prototypical, indirect measurement process in high energy astrophysics. The physical properties of interest are occluded from direct measurement and have to be reconstructed by the analysis techniques from indirect observations. Despite employing state-of-the-art techniques, these methods still suffer from limited acceptance (not all events are detected), limited resolution (not all detected events are reconstructed perfectly) and an irreducible background.

7.1 Inverse Problems

Any measurement process can be described mathematically as folding the probability density function of the property of interest $f(x)$ with the detector response $A(x, y)$ that transforms the true random variable x into the observed quantity y and adding a background $b(y)$ to get the probability density of the observed quantity $g(y)$ [[37](#), Chapter 11].

$$g(y) = \int A(x, y)f(x) dx + b(y) \quad (7.1)$$

This is a Fredholm integral equation of the second kind and the formulation of the inverse problem of finding $f(x)$ from observations $g(y)$. An in-depth treatment about modern approaches to dealing with inverse problems is available in [[31](#)].

There are two possibilities of treating limited acceptance: first A includes the limit acceptance so $\iint A(x, y)f(x) dx dy < 1$ or treating (7.1) only for the detectable events and applying the acceptance correction after an estimation for $f(x)$ has been found. In gamma-ray astronomy, usually the second path is chosen where solving (7.1) treats the limited energy resolution and the acceptance correction is performed using the effective collection area.

To find a solution for (7.1), the continuous version dealing with probability density functions is discretized in the observed and searched-for quantity, transforming it into the matrix equation

$$\mathbf{g} = \mathbf{A} \cdot \mathbf{f} + \mathbf{b} \quad (7.2)$$

where \mathbf{g} and \mathbf{b} are M -vectors containing the event counts of each bin of the observed quantity, \mathbf{f} is an N -vector containing the expected counts in each bin of the searched-for quantity and \mathbf{A} is an $M \times N$ matrix modeling the detector response. The particle flux differential in energy, time and

area is then

$$\Phi_i = \frac{\hat{f}_i}{\Delta E_i \cdot A_{\text{eff},i} \cdot t_{\text{obs}}}. \quad (7.3)$$

Where \hat{f}_i is the estimated solution of the inverse problem, ΔE_i is the width of the energy interval i , $A_{\text{eff},i}$ is the effective area in that interval and t_{obs} is the observation time.

Unfortunately, the naive solution of subtracting \mathbf{b} and left-multiplying the inverse of \mathbf{A} often leads to unacceptable solutions with large bin-wise oscillations due to the bad condition of \mathbf{A} typical in inverse problems. [37, Chapter 11]

Two possible approaches can be made for the parameterization of f . For testing theoretical models or providing parameterizations only using few parameters, a continuous version of $f(x, \theta)$ depending on some parameter vector θ is integrated in certain intervals to obtain discrete predictions for f . This approach is usually referred to as *forward folding* and was used by the MAGIC collaboration to obtain the log-parabola parameterization (1.4) used as reference spectrum for this work thus far [22]. Another approach with the advantage of being free of assumptions about the spectral shape is treating all entries of f as free parameters, describing the true quantity as an unconstrained step function. This is called unfolding or deconvolution.

7.2 Poisson-Maximum-Likelihood Unfolding

Several different algorithms exist for one-dimensional unfolding and have been found to be equally capable of solving the inverse problems in astrophysics [51]. As \mathbf{g} is a histogram, containing integer counts of events, its entries are expected to follow Poisson distributions with the expected value of each bin given by the right hand side of (7.2). Under this assumption, it is possible to compute the likelihood function for observing a given vector \mathbf{g} as

$$\mathcal{L} = \prod_{i=0}^N \frac{\lambda_i^{g_i}}{g_i!} e^{-\lambda_i} \quad (7.4)$$

with

$$\lambda_i = (\mathbf{A} \cdot \mathbf{f} + \mathbf{b})_i. \quad (7.5)$$

For finding numerical solutions to (7.4), it is usually more feasible to minimize or sample the negative logarithm of it. With constant terms under differentiation discarded this yields:

$$-\log \mathcal{L} = \sum_{i=0}^N -g_i \log \lambda_i + \lambda_i. \quad (7.6)$$

It is often required in inverse problems to suppress oscillating, non-sensical solutions that arise from the ill-posedness of the inverse problem by introducing additional information from prior knowledge, e. g. by adding a term to (7.6) punishing large or non-flat solutions. One approach to this is called Tikhonov regularization and results in the following modified negative log-likelihood:

$$-\log \mathcal{L} = \sum_{i=0}^N -g_i \log \lambda_i + \lambda_i - \frac{1}{2} (\mathbf{C}\hat{\mathbf{f}})^\top (\tau \mathbf{1})^{-1} (\mathbf{C}\hat{\mathbf{f}}). \quad (7.7)$$

Here, \mathbf{C} is the Tikhonov regularization matrix. In astrophysics, where power-law or similar spectra are expected, it is usually chosen so that it calculates the discrete second order derivative, for

$N = 4$:

$$C = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 1 & -1 \end{pmatrix} \quad (7.8)$$

It is also important to note, that the assumption of flatness only holds for the logarithm of the acceptance corrected \hat{f} , not for the expected number of events, as these are limited by acceptance and will usually follow more of a bell curve. So for application in astrophysics, \hat{f}_i is replaced with $\log(\hat{f}_i/A_{\text{eff},i})$ in the regularization term. [39, Chapter 4]

In the last years, Bayesian approaches have peaked interest [31, Section 2]. The posterior likelihood can be sampled using Markov chain Monte Carlo (MCMC) providing a fuller picture of the results than just an estimator obtained from numerically minimizing the negative log-likelihood and its covariance from the Fisher information. An efficient MCMC sampling method, mainly characterized via short auto-correlation times, is described in [80] and implemented in the Python package `emcee`. This was used for unfolding of the atmospheric neutrino spectrum as observed by IceCube in [39], implemented in the `funfolding` [40] library.

7.2.1 Application to FACT measurements

A program to unfold event lists as measured by FACT using `funfolding` has been developed as part of this thesis [122]. It reads reconstructed, simulated gamma-rays and information about all simulated events to calculate effective area A_{eff} and the energy migration matrix A . The discretization is done in intervals of estimated energy E_{est} that are equidistant in the logarithm of the energy. The background is estimated from the off positions and assumed to be exact.

The results of applying the regularized unfolding to the Crab Nebula dataset are shown in Figure 7.1, using each of the three different simulation sets for constructing the response matrix A and the effective area A_{eff} . The configuration used is shown in Appendix 17.

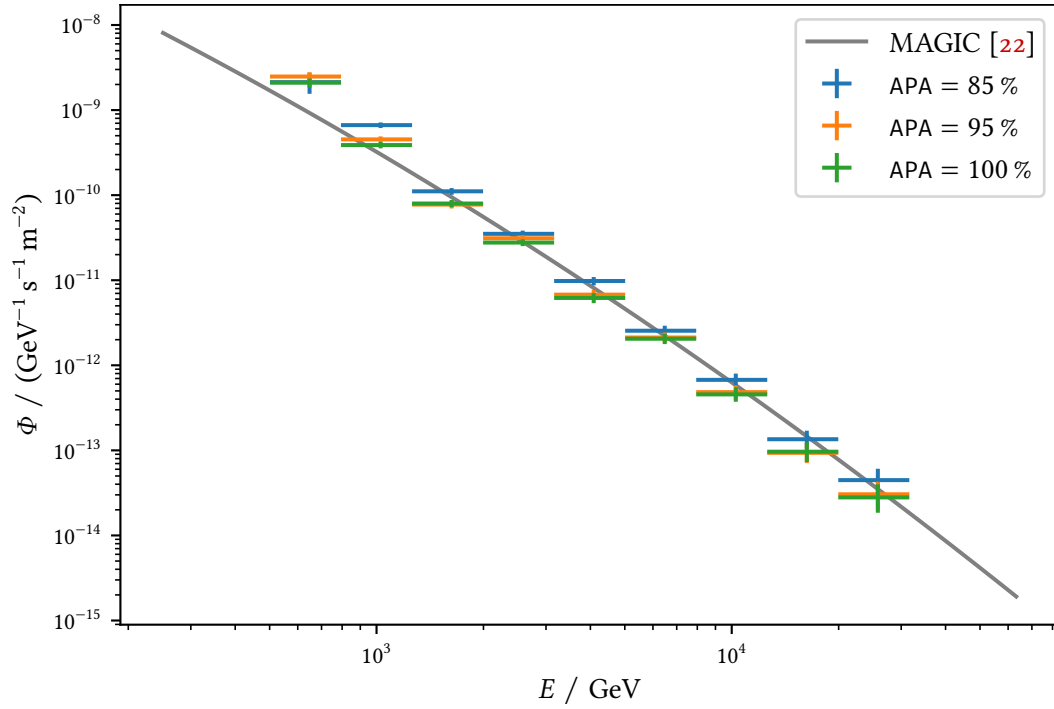


Figure 7.1: Comparison of the spectra unfolded using the different simulated datasets for the IRF calculation. Unfolding using the APA = 85 % dataset fits best with the spectrum published by the MAGIC collaboration for energies larger than 1 TeV while it shows significant disagreement at lower energies. Agreement at low energies is better for the other two datasets but worse at higher energies.

Part II

Automatization of the First G-APD Cherenkov Telescope

Robotic Operation of FACT

From the very beginning, it was planned to operate FACT remotely, to spare expenses and human resources, since FACT is just a small experiment contributed to by four institutes. For a few months after the first light in October 2011, local operators steered the telescope from the container right next to the telescope. Tasks were further automated and FACT could be operated fully remotely by July 2012 [25]. Further steps to minimize the amount of work the operators had to do were made to the point where the telescope operated automatically, without human interaction, after the startup each night. Now, the operators only duty was to stay awake, monitoring the system and weather conditions, only acting when something went wrong. Error conditions include systems failure, wind or clouds coming up. This developed to be a strenuous task, especially in nights with perfect conditions operators were awake without ever needing to do anything. So the question arose, if it was possible to implement a system allowing the operators to sleep and only be alarmed if human interaction became necessary. This system called the `shiftheLper` was developed and tested in several iterations, mainly by Dominik Neise and me. After a lengthy period of tests, the FACT collaboration decided the system was ready and operators could sleep at night at the end of 2017.

In this chapter, I will discuss the automatization and remote control infrastructure of FACT and how the `shiftheLper` was build on top of these existing systems to enable robotic operation.

8.1 Remote Control Infrastructure

FACT comprises many subsystems that need to work in unison for observations. The different subsystems provide information about the telescope's status, weather conditions and most importantly steer the telescope, configure the telescope's hardware and take data. The communication is performed over an inter-process communication system based on the Distributed Information Management System (DIM) [79] library. Each subsystem is a DIM service, which can implement status updates and/or commands. DIM-clients can request status updates at regular intervals or on changes and can issue commands.

The telescope's operation is steered using JavaScript and a webinterface called `SmartFact` provides the current status of the telescope, weather information, the possibility to manually control the drive and the ability to start operation scripts. A screenshot of the front page is shown in [Figure 8.1](#).

The observation steering scripts are interpreted on the FACT web server and write commands into the DIM network and receive status updates from it. During the night, the script `Main.js` runs a loop getting the observation schedule from a database and executing the scheduled measurements by sending the appropriate DIM commands. Operators are assigned to observation nights using a calendar application and while there is a webinterface for the observation schedule, this is also filled automatically nowadays. As an example, the schedule for December 19th, 2019 is shown in

Table 8.1. This night, observations for five sources were carried out and the shift was eleven and a half hours long.

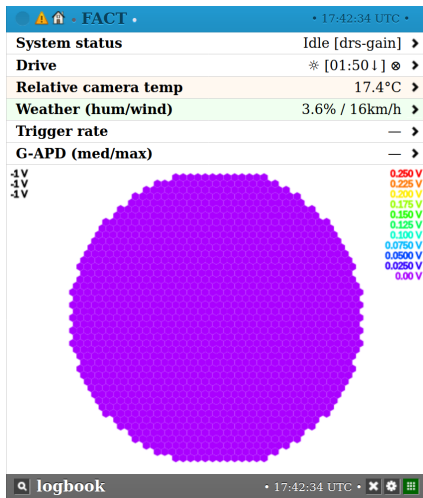


Figure 8.1: Homepage of SmartFact, the webinterface for the remote operation of FACT during day time.

Startup in the evening is organized via working through a checklist, which includes—among other things—inspecting the telescope via a live camera feed, powering the drive electronics, and starting execution of the `Main.js` script.

During the night, the system and weather conditions have to be monitored. This was the task of the operator, who needed to stay awake for the whole preparation and observation time, which is over eleven hours in winter.

The most critical task is the shutdown of the telescope in the morning, as the telescope is a serious fire hazard [71], if not in its parking position due North during daylight. After the shutdown procedure is executed, the operator goes through a checklist again and the result is stored in the FACT database. Before the `shiftheper` was deployed, not filling this checklist only resulted in an email to the FACT collaboration, easily missed completely or seen to late to react in time.

8.2 The shiftheper

To move to fully automated data taking without a human operator needed to keep an eye on system status and weather conditions, the `shiftheper`¹ has been developed. As a continuously running service, the `shiftheper` performs all checks previously done manually by the operators. Additional checks were added to make sure the observations are started in the evening, the telescope is parked in the morning, possible alerts have indeed reached a human operator, and the `shiftheper` itself is running.

Table 8.1: Schedule for 2019-12-19.

Time	Type	Source
19:03	Startup	—
19:18	Data	1ES 1959+650
21:22	Data	1ES 2344+51.4
00:35	Data	Crab
04:03	Data	PKS 0736+01
05:21	Data	Mrk 421
06:22	Shutdown	—

Twilio [150], by issuing http post requests to a web page or just logging messages to file or console. The `shiftheper` makes use of the instant messages, phone calls, website and logging facilities.

¹<https://github.com/fact-project/shiftheper>

The severity level is represented as an integer, comparable to common logging libraries. Notifiers have a level setting under which messages will not be relayed to the recipients and messages can be relayed to different recipients depending on the category of a message. This is used in the `shiftheelper` to call either the current operator for problems directly related to the observations or an expert in case of problems with the `shiftheelper` itself or if no operator is available. For notifiers that support it, e. g. email and Telegram, files and images can be attached.

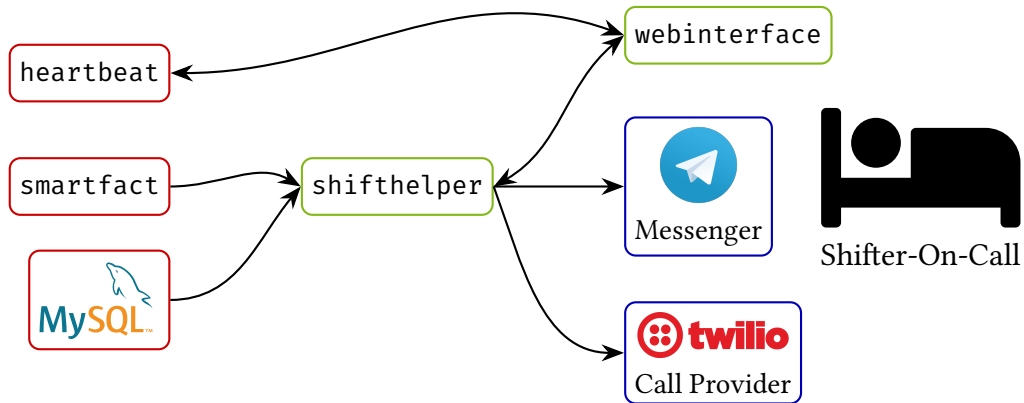


Figure 8.2: Structure of the `shiftheelper` infrastructure. The services in red are running on the computers in the counting container on La Palma, the ones in green are running on a server at TU Dortmund and the blue ones are external services. The arrows show the flow directions of information.

The `shiftheelper` is running on a server in continental Europe, either in Dortmund or at ETH Zurich, so that it can still alert the operator even if power was lost at the telescope site or internet access at the observatory is not possible. The current operator is queried from the FACT database running at the observatory, while a fallback operator and the expert are set for longer periods in a configuration file to also work in case no internet connection to La Palma is possible. To be safe for short losses of connection, the parts of the database needed by the `shiftheelper` are cached on the server the `shiftheelper` is running on and synced with the database on La Palma every minute.

The checks currently performed by the `shiftheelper` are listed in [Table 8.2](#) together with who is called in case the check fails. Most of these checks are only performed during planned observations, defined as ten minutes before the scheduled startup and the scheduled shutdown. A lower level of checks makes sure the information provided through `SmartFact` is up to date, e. g. `SmartFact` is running and weather information has been updated in the last ten minutes. Checks monitoring the environmental conditions are using two weather stations on the Roque not far from FACT, provided by MAGIC and the Telescopio Nazionale Galileo, the data is exposed through `SmartFact`. Currently, only wind speed is monitored as rain is considered not harmful for the telescope. During observations with strong moonlight, it can happen that the SIPM current exceeds safety limits. Two limits are defined and checked by the `shiftheelper`, one for the median current over all pixels and one for the maximum camera current. During the night, it is checked if a script is running, so that when `Main.js` exits due to an error or was never started in the beginning of the night, the operator is notified. Several checks monitor the state of different subsystems of FACT, including

Table 8.2: Checks performed by the `shiftheelper`. Most checks are only performed during scheduled data taking, between 10 minutes before the scheduled startup and the shutdown task. Only the check if the operator performed the shutdown is done after the scheduled shutdown time.

Condition	Callee
Is the SmartFact website up to date?	Operator
Is weather information available?	Operator
Is an operator in the calendar for the night?	Expert
Is <code>Main.js</code> running?	Operator
Is the wind speed slower than 50 km/h if the telescope is not parked	Operator
Is the wind gust speed slower than 50 km/h if the telescope is not parked	Operator
Is the median camera current less than 115 μA	Operator
Is the maximum camera current less than 160 μA	Operator
Is the camera cooling working?	Operator
Low trigger rate for longer than ten minutes?	Operator
Is the temperatures of the container nominal?	Operator
Is the operator awake 20 minutes before shutdown?	Operator
Is the parking checklist filled after shutdown?	Operator
Is the <code>heartbeat</code> program running?	Expert
Two checks if the DIM network is operating correctly	Operator
Three different checks for correct operation of the bias voltage supply	Operator
A dummy alert, can be triggered by the operator to test the configuration	Operator

the bias voltage supply for the pixels. Most importantly, it is checked if the human operator is awake to supervise the shutdown procedure and that the shutdown checklist has been filled.

Exceptions occurring while the `shiftheLper` performs its checks result in a message to the expert, notifying him or her of the error condition. Through this very general instrument, the `shiftheLper` identified several unanticipated error conditions and immediately notified the collaboration. This includes network outages on La Palma, power losses in the observatory and a hardware failure.

Since first testing began in 2016, costs for phone calls have cumulated at around 350 €. Operators should normally not answer the phone calls to reduce costs, and most calls are indeed free of charge. Large parts of the costs were generated by few operators sending calls to their mailboxes, something now easily avoided by making operators aware of the issue. Compared to the cost of human labor, this is a rather small amount of money for nearly four years of operation².

8.3 Webinterface

To display the current alerts and to make sure the operator has received and acknowledged a message, a web service has been developed. It receives messages from the `shiftheLper` via http POST requests and displays a list of alerts. Operators have to acknowledge the alerts by pressing a button. If an alert is not acknowledged within 15 minutes, the fallback operator is called to take over. Acknowledging an alert will also prevent the `shiftheLper` for calling the operator for the same reason for a time period of ten minutes, giving the operator time to check what has happened and probably solving the issue. A screenshot of the webinterface is shown in [Figure 8.3](#). The webinterface uses the WebSocket protocol to push new alerts to the connected clients, so they directly appear without refreshing the page.

The webinterface is also used to check if the human operator is awake twenty minutes before the scheduled shutdown. For this, the operator has to push a button on the webinterface.

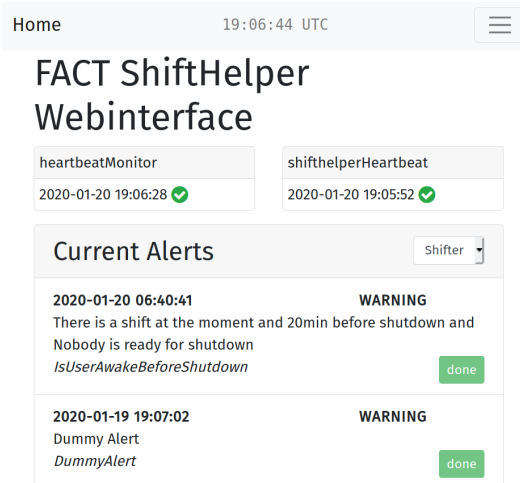


Figure 8.3: Screenshot of the `shiftheLper` homepage, showing the most recently received heartbeats and the list of alerts. The dummy alert is part of the startup checklist and is issued by the operator via the webinterface to make sure the `shiftheLper` is running and has the correct contact information. The webinterface is optimized for use on small screens such as mobile phones.

²FACT operators were not paid directly for the night shifts. Rather it was expected of collaboration members to perform these shifts. However, the night shifts took a toll on the operators and usually reduced the ability to work on their day job, e. g. as research assistant

8.4 Heartbeat

To make sure the `shiftheper` is up and running, it posts a heartbeat timestamp every minute to the webinterface which is checked by a program running on La Palma. In case the timestamp is not available or older than 10 minutes, the heartbeat program calls the expert. In turn, the heartbeat program also posts a timestamp to the webinterface which is checked by the `shiftheper` itself, thus providing strong guarantees that both programs are running at all times minimizing the chances of unnoticed failure.

Conclusions and Outlook

In the course of this thesis, I presented the work performed to improve the capabilities of the FACT-Tools data analysis pipeline from the necessary simulations to the event reconstruction and the operation of the FACT telescope itself.

For the simulations, several large improvements were made. Switching to the most recent version of CORSIKA available at the time resulted in much faster computation and inclusion of the latest models for hadronic interactions. With the implementation of mopro in version 3, FACT has the means to produce the necessary amounts of simulations as quick as possible in a reproducible and nearly completely automatized manner. This enabled producing enough proton statistics to be able to calculate spectral sensitivity on simulated data for the first time, which was computationally infeasible before. It also dramatically reduced the effort necessary to simulate different sets of simulations. However, the feedback loop for changes is still very slow and the main reason, why even after eight years of operation, agreement of simulations and observed data is still the most limiting issue at the lower energies. To better narrow down the possible reasons, cross checking the results of the detector simulation with other programs than ceres, such as sim_telarray [33], might give valuable insights on where to search for the remaining disagreements. Compared to the older version of mopro, it is also possible to easily run simulations using multiple compile-time configurations of CORSIKA. This was used to make the decision to drop FLUKA and use URQMD for the low energy hadronic interactions.

The main result of my thesis, achieving a gain of 45 % in integral sensitivity over previous works, is the sum of many minor changes to the analysis software over the course of almost four years but is dominated by the introduction of a new, machine learning based reconstruction of the particle origin via the disp method. The implemented method improves over previous approaches by fully utilizing machine learning in the two step procedure of estimating absolute value and sign of disp. Background suppression and angular resolution are improved dramatically at the cost of reduced effective area at the lowest energies.

The new method for the reconstruction of gamma-ray origin together with the elimination of features dependent on a known point-source position enabled the first creation of skymaps of FACT data. A smaller dataset of events reconstructed using this improved analysis and the corresponding instrument response functions were made public and then used to publish the first multi-instrument analysis based on open data by four of the currently operating Cherenkov telescopes in [121].

Further improvements can be made to this analysis chain, most crucially through the improvement of agreement between observations and simulations. A very promising approach here is the addition of measured electronic and night sky background noise to simulated data [52, 50]. As soon as this is achieved, more can be gained by optimizing the hyperparameters of the reconstruction models, as the ones used in this work are just a *working* solution but no exhaustive search for an optimal set of parameters has been performed. As the quality metrics of background suppression

and origin reconstruction are highly energy dependent, most telescopes employ energy dependent event selection criteria, usually optimized for the best sensitivity in each interval of estimated energy. This approach is also worth pursuing for the analysis presented here and could improve acceptance and sensitivity especially in the lowest energies. Deep learning approaches have shown promising results for other telescopes [141] and might be able to outperform the *classical* event reconstruction once the necessary simulation agreement has been reached.

For the unfolding, the full potential of the Bayesian likelihood-based approach using MCMC-sampling has not yet been employed. In [39], also systematic uncertainties—like certain not perfectly known detector properties—were modeled in the likelihood. This could also be used for FACT, e. g. to model overall light detection efficiency. Another extension of the likelihood is also treating the background as Poisson distributed, as done in [47].

Lessons learned during the development of FACT-Tools have already been applied and contributed into the analysis software `ctapipe` [103, 102] for the upcoming Cherenkov Telescope Array, such as the calculation of the Hillas parameters based on a principal component analysis and several implementations of other image features. I also contributed a major part of the coordinate system transformations in `ctapipe`, which only was possible through the experience gained with the work on FACT. The python library to read the event `io` output of `CORSIKA`, first implemented to read information about the simulated `mopro` events, was extended by Dominik Neise and me to also be able to read the output of `simtel_array` and it is now used to read the CTA simulations into `ctapipe`. The `aict-tools`, including the reconstruction of origin based on the `disp` method implemented for this thesis, is currently being for reconstruction of the first data taken by the CHEC [140] camera prototype for the Small Sized Telescope of CTA [158].

The `shiftheelper` had a large impact on everyone in the FACT collaboration. It made observations under nominal conditions nearly effortless, increased the duty cycle by directly alerting the operators in case human intervention was necessary and it improved the safety of the instrument by continuously checking for adverse conditions and making sure the telescope is parked in the morning. Further automatizations are planned, e. g. automatically parking the telescope in case of strong wind, instead of calling an operator to perform this task.

In the last several years, focus in the scientific community and especially in particle and astroparticle physics has shifted away from proprietary tools and limited access to observed data towards the recognition, that open source software, publicly available data and open access publications are a bare necessity for successful and reproducible science. Gamma-ray astronomy has lagged behind quite a bit in this regard, but with the advent of CTA, which is to be operated as open observatory, this is changing. In this light, a major focus of this work has been placed on reproducibility and basing the analysis on publicly available, widely-used, battle-tested tools. With unfortunate exception of `CORSIKA`, the entire analysis is based on Free and Open Source Software (FOSS) and in general, the necessity for human interaction at every step of the way was minimized as much as possible. The analysis results are stored in standardized data formats, widely used even outside the astroparticle physics community. Together with the final steps described in [47] it is now possible to publish FACT data in the Open Data Format for Gamma-Ray Astronomy [58], enabling joint analysis of multi-instrument data for the first time, as we did in [121]. This document with all included graphics can be created from the input data, configuration and text files by issuing a single command after installing the necessary software. Detailed instructions can be found in Appendix 1.

Bibliography

For closed access publications, also an open access preprint, usually from the [arXiv](#), is provided. Where an open access version could not be found, this is noted.

1. M. G. Aartsen et al. (IceCube Collaboration). “Evidence for High-Energy Extraterrestrial Neutrinos at the IceCube Detector”. *Science* 342:1242856, 6161 2013.
DOI: [10.1126/science.1242856](#). ARXIV: [1311.5238](#) [astro-ph.HE]
2. M. Aartsen, M. Ackermann, et al. (IceCube, Fermi-LAT, MAGIC, AGILE, ASAS-SN, HAWC, H.E.S.S., INTEGRAL, Kanata, Kiso, Kapteyn, Liverpool Telescope, Subaru, Swift/NuSTAR, VERITAS and VLA/17B-403 Collaboration). “Multimessenger observations of a flaring blazar coincident with high-energy neutrino IceCube-170922A”. *Science* 361:6398, 2018.
DOI: [10.1126/science.aat1378](#). ARXIV: [1807.08816](#) [astro-ph.HE]
3. M. Aartsen, M. Ackermann, J. Adams, et al. (IceCube Collaboration). “Neutrino emission from the direction of the blazar TXS 0506+056 prior to the IceCube-170922A alert”. *Science* 361:6398, 2018, pages 147–151.
DOI: [10.1126/science.aat2890](#)
4. B. P. Abbot et al. “Multi-messenger Observations of a Binary Neutron Star Merger”. *The Astrophysical Journal* 848:2, 2017, page L12.
DOI: [10.3847/2041-8213/aa91c9](#)
5. B. P. Abbott et al. “Observation of Gravitational Waves from a Binary Black Hole Merger”. *Physical Review Letters* 116:6, 2016. ISSN: 1079-7114.
DOI: [10.1103/physrevlett.116.061102](#)
6. B. P. Abbott et al. “GW170817: observation of gravitational waves from a binary neutron star inspiral”. *Physical Review Letters* 119:16, 2017, page 161101
7. H. Abdalla et al. “The H.E.S.S. Galactic plane survey”. *Astronomy & Astrophysics* 612, 2018, A1.
DOI: [10.1051/0004-6361/201732098](#)
8. S. Abdollahi, F. Acero, M. Ackermann, et al. (Fermi-LAT Collaboration). *Fermi Large Area Telescope Fourth Source Catalog*. 2019.
ARXIV: [1902.10045](#) [astro-ph.HE]
9. A. U. Abeysekara et al. “Daily Monitoring of TeV Gamma-Ray Emission from Mrk 421, Mrk 501, and the Crab Nebula with HAWC”. *The Astrophysical Journal* 841:2, 2017, page 100.
DOI: [10.3847/1538-4357/aa729e](#)
10. V. A. Acciari, S. Ansoldi, L. Antonelli, et al. (MAGIC Collaboration). “Teraelectronvolt emission from the γ -ray burst GRB 190114C”. *Nature* 575, 2019, page 455

Bibliography

11. B. S. Acharya et al. (CTA Collaboration). *Science with the Cherenkov Telescope Array*. World Scientific, 2017
12. A. Achterberg et al. “First year performance of the IceCube neutrino telescope”. *Astroparticle Physics* 26:3, 2006, pages 155–173.
DOI: [10.1016/j.astropartphys.2006.06.007](https://doi.org/10.1016/j.astropartphys.2006.06.007). ARXIV: [astro-ph/0604450](https://arxiv.org/abs/astro-ph/0604450) [astro-ph]
13. F. Aharonian et al. “The Crab nebula and pulsar between 500 GeV and 80 TeV: observations with the HEGRA stereoscopic air Cerenkov telescopes”. *The Astrophysical Journal* 614:2, 2004, page 897.
DOI: [10.1086/423931](https://doi.org/10.1086/423931)
14. F. Aharonian et al. “Observations of the Crab nebula with HESS”. *Astronomy & Astrophysics* 457:3, 2006, pages 899–915.
DOI: [10.1051/0004-6361:20065351](https://doi.org/10.1051/0004-6361:20065351)
15. M. L. Ahnen et al. “Data compression for the first G-APD Cherenkov Telescope”. *Astronomy and Computing* 12, 2015, pages 191–199.
DOI: [10.1016/j.ascom.2015.06.007](https://doi.org/10.1016/j.ascom.2015.06.007). ARXIV: [1506.06045](https://arxiv.org/abs/1506.06045) [astro-ph.IM]
16. M. L. Ahnen et al. “Normalized and Asynchronous Mirror Alignment for Cherenkov Telescopes”. *Astroparticle Physics* 82, 2016, pages 56–65.
DOI: [10.1016/j.astropartphys.2016.05.005](https://doi.org/10.1016/j.astropartphys.2016.05.005). ARXIV: [1605.06274v1](https://arxiv.org/abs/1605.06274v1) [astro-ph.IM]
17. M. Ajello et al. “3FHL: The Third Catalog of Hard Fermi-LAT Sources”. *The Astrophysical Journal Supplement Series* 232:2, 2017.
DOI: [10.3847/1538-4365/aa8221](https://doi.org/10.3847/1538-4365/aa8221)
18. K. Akiyama et al. “First M87 Event Horizon Telescope Results. IV. Imaging the Central Supermassive Black Hole”. *The Astrophysical Journal Letters* 875:1, 2019, page L4.
DOI: [10.3847/2041-8213/ab0e85](https://doi.org/10.3847/2041-8213/ab0e85)
19. J. Albert et al. (MAGIC Collaboration). “Variable Very High Energy γ -Ray Emission from Markarian 501”. *The Astrophysical Journal* 669:2, 2007, pages 862–883.
DOI: [10.1086/521382](https://doi.org/10.1086/521382)
20. J. Albert et al. “VHE γ -Ray Observation of the Crab Nebula and its Pulsar with the MAGIC Telescope”. *The Astrophysical Journal* 674:2, 2008, pages 1037–1055.
DOI: [10.1086/525270](https://doi.org/10.1086/525270). ARXIV: [0705.3244](https://arxiv.org/abs/0705.3244) [astro-ph]
21. J. Aleksić et al. “The major upgrade of the MAGIC telescopes, Part II: A performance study using observations of the Crab Nebula”. *Astroparticle Physics* 72, 2016, pages 76–94.
DOI: [10.1016/j.astropartphys.2015.02.005](https://doi.org/10.1016/j.astropartphys.2015.02.005). ARXIV: [1409.5594](https://arxiv.org/abs/1409.5594) [astro-ph.IM]
22. J. Aleksić et al. (MAGIC Collaboration). “Measurement of the Crab Nebula spectrum over three decades in energy with the MAGIC telescopes”. *Journal of High Energy Astrophysics* 5, 2015, pages 30–38.
DOI: [10.1016/j.jheap.2015.01.002](https://doi.org/10.1016/j.jheap.2015.01.002). ARXIV: [1406.6892](https://arxiv.org/abs/1406.6892) [astro-ph.HE]

23. *AMON IceCube HESE Event Information*. 2019.
https://gcn.gsfc.nasa.gov/amon_hese_events.html visited on 2019-11-18
24. *An Introduction to the Silicon Photomultiplier*. Technical report Rev. 6.0. sensL, 2017
25. H. Anderhub et al. (FACT Collaboration). “Design and operation of FACT – the first G-APD Cherenkov telescope”. *Journal of Instrumentation* 8:06, 2013.
DOI: 10.1088/1748-0221/8/06/p06008. ARXIV: 1304.1710 [astro-ph.IM]
26. J. W. M. Baars and A. P. Hartsuijker. “The Decrease of Flux Density of Cassiopeia A and the Absolute Spectra of Cassiopeia A, Cygnus A and Taurus A”. *Astronomy & Astrophysics* 17, 1972, page 172.
<https://ui.adsabs.harvard.edu/abs/1972A%26A...17..172B/abstract>
27. Bandiera, R., Neri, R., and Cesaroni, R. “The Crab Nebula at 1.3 mm - Evidence for a new synchrotron component”. *Astronomy & Astrophysics* 386:3, 2002, pages 1044–1054.
DOI: 10.1051/0004-6361:20020325
28. S. Bass. “Microscopic models for ultrarelativistic heavy ion collisions”. *Progress in Particle and Nuclear Physics* 41, 1998, pages 255–369.
DOI: 10.1016/s0146-6410(98)00058-1. ARXIV: nucl-th/9803035
29. O. Ben-Kiki, C. Ecanas, and I. döt Net. *YAML Ain't Markup Language (YAML™) Version 1.2*. 3rd ed. 2009
30. W. Benbow et al. “Direct measurement of stellar angular diameters by the VERITAS Cherenkov telescopes”. *Nature Astronomy* 3:6, 2019, pages 511–516.
DOI: 10.1038/s41550-019-0741-z. ARXIV: 1904.06324 [astro-ph.SR]
31. M. Benning and M. Burger. “Modern regularization methods for inverse problems”. *Acta Numerica* 27, 2018, pages 1–111.
DOI: 10.1017/s0962492918000016
32. K. Bernlöhr. *eventio. a machine-independent hierarchical data format and its programming interface*. 2014.
https://www.mpi-hd.mpg.de/hfm/~bernlöhr/sim_telarray/Documentation/eventio_en.pdf
33. K. Bernlöhr. “Simulation of imaging atmospheric Cherenkov telescopes with CORSIKA and sim_telarray”. *Astroparticle Physics* 30:3, 2008, pages 149–158.
DOI: 10.1016/j.astropartphys.2008.07.009. ARXIV: 0808.2253 [astro-ph]
34. A. F. Bielajew et al. *History, overview and recent improvements of EGS4*. Technical report SLAC-PUB-6499. SLAC, 1994
35. A. Biland et al. (FACT Collaboration). “Calibration and performance of the photon sensor response of FACT – the first G-APD Cherenkov telescope”. *Journal of Instrumentation* 9:10, 2014.
DOI: 10.1088/1748-0221/9/10/P10012. ARXIV: 1403.5747 [astro-ph.IM]
36. A. Biland (FACT Collaboration). *Upper limit on TeV emission from Fermi J0139.5+2928 measured with FACT*. 2019.
<http://www.astronomerstelegam.org/?read=12891>

Bibliography

37. V. Blobel and E. Lohrmann. *Statistische und numerische Methoden der Datenanalyse*. 2nd ed. 2012.
<http://www.desy.de/~blobel/eBuch.pdf>
38. C. Bockermann. “Mining big data streams for multiple concepts”. PhD thesis. TU Dortmund, 2015.
doi: [10.17877/DE290R-16437](https://doi.org/10.17877/DE290R-16437)
39. M. Börner. “Bestimmung des Energiespektrums von atmosphärischen Myonenneutrinos mit 3 Jahren Daten des IceCube-Detektors”. PhD thesis. TU Dortmund, 2018.
doi: [10.17877/DE290R-19089](https://doi.org/10.17877/DE290R-19089)
40. M. Börner, M. Nöthe, and J. Bieker. *funfolding. Implementation of Likelihood-Based unfolding techniques and Decision-Tree based binning*. Version v0.2.1.
<https://github.com/tudo-astroparticlephysics/funfolding>
41. L. Breiman et al. *Classification and Regression Trees*. Chapman & Hall/CRC, 1984
42. L. Breiman. “Bagging predictors”. *Machine learning* 24:2, 1996, pages 123–140.
doi: [10.1023/A:1018054314350](https://doi.org/10.1023/A:1018054314350)
43. L. Breiman. “Random forests”. *Machine learning* 45:1, 2001, pages 5–32.
doi: [10.1023/A:1010933404324](https://doi.org/10.1023/A:1010933404324)
44. T. Bretz. *FACT++ source code repository*. 2019.
<https://trac.fact-project.org/browser/trunk/FACT%2B%2B?rev=19559>
45. T. Bretz and D. Dorner. “MARS-CheObs goes Monte Carlo”. In: *Proceedings, 31th International Cosmic Ray Conference*. Łódź, Poland, 2009, page 1259
46. K. A. Brügge, J. Buß, et al. “Real Time Streaming Analysis of IACT Data”. In: *Proceedings, Astronomical Data Analysis Software and Systems XXVI*. Vol. 521. 2016, page 335
47. K. A. Brügge. “Unmasking The Gamma-Ray Sky. Comprehensive and Reproducible Analysis for Cherenkov Telescopes”. PhD thesis. TU Dortmund, 2019.
doi: [10.17877/DE290R-20440](https://doi.org/10.17877/DE290R-20440)
48. K. A. Brügge et al. *FACT-Tools*. Version v1.1.3. 2019.
doi: [10.5281/zenodo.3529353](https://doi.org/10.5281/zenodo.3529353). <https://github.com/fact-project/fact-tools>
49. R. Buehler et al. “Gamma-Ray Activity in the Crab Nebula: the Exceptional Flare of April 2011”. *The Astrophysical Journal* 749:1, 2012, page 26.
doi: [10.1088/0004-637x/749/1/26](https://doi.org/10.1088/0004-637x/749/1/26). ARXIV: [1112.1979](https://arxiv.org/abs/1112.1979) [astro-ph.HE]
50. M. Bulinski. “OBASS – Observed Background and Artificial Signal Superposition. Entwicklung und Evaluation einer Methode zur Verbesserung der Simulation für FACT mittels Superposition von gemessenem Nachthimmel und simulierten Luftschauern.” German. Master’s thesis. TU Dortmund, 2018
51. M. Bunse et al. “Unification of deconvolution algorithms for cherenkov astronomy”. In: *Proceedings, 5th International Conference on Data Science and Advanced Analytics*. IEEE. 2018, pages 21–30

52. J. B. Buß. “FACT in the Bright Light of the Moon. Extending Observations with IACTs to Full Moon Nights by using SiPMs”. PhD thesis. TU Dortmund, 2020.
Note: Unpublished work at time of writing
53. CTA Consortium. *Current progress toward construction and the first telescopes on site*. 2019.
<https://www.cta-observatory.org/project/status> visited on 2019-11-12
54. CTA Consortium. *The LST-1 Detects its First Gamma-Ray Signal*. 2019.
<https://www.cta-observatory.org/lst1-detects-first-gamma-ray-signal> visited on 2019-12-11
55. J. M. Davies and E. S. Cotton. “Design of the quartermaster solar furnace”. *Solar Energy* 1:2-3, 1957, pages 16–22.
DOI: [10.1016/0038-092X\(57\)90116-0](https://doi.org/10.1016/0038-092X(57)90116-0).
Note: No open access available
56. A. De Angelis and M. Mallamaci. “Gamma-ray Astrophysics”. *The European Physical Journal Plus* 133:8, 2018.
DOI: [10.1140/epjp/i2018-12181-0](https://doi.org/10.1140/epjp/i2018-12181-0)
57. A. De Angelis and M. Pimenta. *Introduction to particle and astroparticle physics. Multimessenger astronomy and its particle physics foundations*. Springer, 2018
58. C. Deil et al. *Data formats for gamma-ray astronomy - version 0.2*. 2018.
DOI: [10.5281/zenodo.1409831](https://doi.org/10.5281/zenodo.1409831). <https://gamma-astro-data-formats.readthedocs.io/en/latest>
59. C. Deil et al. “Open high-level data formats and software for gamma-ray astronomy”. In: *Proceedings, 6th International Meeting on High Energy Gamma-Ray Astronomy*. 2016.
DOI: [10.1063/1.4969003](https://doi.org/10.1063/1.4969003). ARXIV: [1610.01884](https://arxiv.org/abs/1610.01884) [astro-ph.IM]
60. P. Doll et al. *The Karlsruhe Cosmic Ray Project KASCADE*. Technical report KfK4686. Kernforschungszentrum Karlsruhe, 1990
61. E. Domingo-Santamaria et al. (MAGIC Collaboration). “The DISP analysis method for point-like or extended gamma source searches / studies with the MAGIC telescope”. In: *Proceedings, 29th International Cosmic Ray Conference*. Puna, India, 2005.
ARXIV: [astro-ph/0508274](https://arxiv.org/abs/astro-ph/0508274) [astro-ph]
62. D. Dorner et al. (FACT Collaboration). “FACT – Monitoring Blazars at Very High Energies”. In: *Proceedings, 5th International Fermi Symposium*. 2015.
ARXIV: [1502.02582](https://arxiv.org/abs/1502.02582) [astro-ph.IM]
63. D. Dorner et al. (FACT Collaboration). “FACT – Highlights from more than Seven Years of Unbiased Monitoring at TeV Energies”. In: *Proceedings, 36th International Cosmic Ray Conference*. 665. Madison, USAa, 2019.
<https://pos.sissa.it/358/665>
64. H.-J. Drescher et al. “Model dependence of lateral distribution functions of high energy cosmic ray air showers”. *Astroparticle Physics* 21:1, 2004, pages 87–94.
DOI: [10.1016/j.astropartphys.2003.10.007](https://doi.org/10.1016/j.astropartphys.2003.10.007). ARXIV: [astro-ph/0307453](https://arxiv.org/abs/astro-ph/0307453)

Bibliography

65. Y. Du and F. Retiere. “After-pulsing and cross-talk in multi-pixel photon counters”. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 596:3, 2008, pages 396–401.
doi: [10.1016/j.nima.2008.08.130](https://doi.org/10.1016/j.nima.2008.08.130)
66. R. Engel et al. “Towards A Next Generation of CORSIKA: A Framework for the Simulation of Particle Cascades in Astroparticle Physics”. *Computing and Software for Big Science* 3:1, 2019, page 2
67. FACT Collaboration. *Subversion Repository*. 2019.
<https://trac.fact-project.org/browser/trunk>
68. *FACT Open Data*. 2017.
<https://fact-project.org/data/> visited on 2019-11-06
69. A. Fasso et al. *FLUKA: a multi-particle transport code*. Technical report R-773. SLAC, 2005.
doi: [10.2172/877507](https://doi.org/10.2172/877507)
70. T. Fawcett. “An introduction to ROC analysis”. *Pattern recognition letters* 27:8, 2006, pages 861–874.
doi: [10.1016/j.patrec.2005.10.010](https://doi.org/10.1016/j.patrec.2005.10.010)
71. *Fire Damages Gamma-Ray Observatory*. 1997.
<https://www.sciencemag.org/news/1997/10/fire-damages-gamma-ray-observatory> visited on 2019-11-25
72. R. Fisher. *Earth Rotation and Equatorial Coordinates*. 2010.
https://www.cv.nrao.edu/~rfisher/Ephemerides/earth_rot.html visited on 2017-11-29
73. FITS Working Group. *Definition of the Flexible Image Transport System(FITS). Version 4.0*. 2016.
https://fits.gsfc.nasa.gov/fits_standard.html
74. FLUKA Team. *FLUKA User License*. 2017.
<http://www.fluka.org/fluka.php?id=license> visited on 2019-11-05
75. M. Folk et al. “An overview of the HDF5 technology suite and its applications”. In: *Proceedings, EDBT/ICDT 2011 Workshop on Array Databases*. 2011, pages 36–47
76. V. P. Fomin et al. “New methods of atmospheric Cherenkov imaging for gamma-ray astronomy. I. The false source method”. *Astroparticle Physics* 2:2, 1994, pages 137–150.
doi: [10.1016/0927-6505\(94\)90036-1](https://doi.org/10.1016/0927-6505(94)90036-1)
77. S. Funk. “Ground- and Space-Based Gamma-Ray Astronomy”. *Annual Review of Nuclear and Particle Science* 65:1, 2015, pages 245–277.
doi: [10.1146/annurev-nucl-102014-022036](https://doi.org/10.1146/annurev-nucl-102014-022036)
78. Gammapy Developers. *Multiwavelength observation of the Crab Nebula*.
https://github.com/gammapy/gammapy-extra/blob/master/datasets/tests/unbundled/tev_spectra/crab_mwl.fits.g://github.com/gammapy/gammapy-extra/blob/23bc4946ca/datasets/tests/unbundled/tev_spectra/crab_mwl.fits.gz visited on 2019-12-12

79. C. Gaspar and M. Dönszelmann. *DIM. A Distributed Information Management System for the DELPHI experiment at CERN*. Technical report. CERN, 1994
80. J. Goodman and J. Weare. “Ensemble samplers with affine invariance”. *Communications in applied mathematics and computational science* 5:1, 2010, pages 65–80
81. D. A. Green, R. Tuffs, and C. Popescu. “Far-infrared and submillimetre observations of the Crab nebula”. *Monthly Notices of the Royal Astronomical Society* 355:4, 2004, pages 1315–1326
82. A. Guazzelli et al. “PMML: An open standard for sharing models”. *The R Journal* 1:1, 2009, pages 60–65.
DOI: [10.32614/RJ-2009-010](https://doi.org/10.32614/RJ-2009-010)
83. T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning. Data mining, Inference and Prediction*. 2nd ed. Springer, 2009.
DOI: [10.1007/b94608](https://doi.org/10.1007/b94608)
84. D. Heck et al. *CORSIKA: A Monte Carlo Code to Simulate Extensive Air Showers*. Technical report FZKA 6019. Forschungszentrum Karlsruhe, 1998
85. D. Heck and T. Pierog. *Extensive Air Shower Simulation with CORSIKA: A User’s Guide. Version 7.6900 from January 17, 2019*. 2019.
www.ikp.kit.edu/corsika/70.php.
Note: Only the most recent version of the documentation can be accessed without password.
86. W. Heitler. *The quantum theory of radiation*. 2nd ed. Oxford University Press, 1944.
<https://archive.org/details/in.ernet.dli.2015.37198>
87. G. S. Hennessy et al. “Ultraviolet imaging telescope observations of the Crab nebula”. *The Astrophysical Journal* 395, 1992, pages L13–L16.
DOI: [10.1086/186477](https://doi.org/10.1086/186477)
88. V. F. Hess. “Über Beobachtungen der durchdringenden Strahlung bei sieben Freiballonfahrten”. *Phys. Zeit* 13, 1912, pages 1084–1091.
<http://inspirehep.net/record/1623161/files/HessArticle.pdf>.
Note: German, English translation available at <https://arxiv.org/abs/1808.02927>.
89. P. Hientjen, M. Sustrik, et al.
<https://zeromq.org>
90. A. M. Hillas. “Cerenkov light images of EAS produced by primary gamma”. In: *Proceedings, 19th International Cosmic Ray Conference*. Vol. 3. 1985
91. J. Hinton (H.E.S.S. Collaboration). “The status of the HESS project”. *New Astronomy Reviews* 48:5-6, 2004, pages 331–337.
DOI: [10.1016/j.newar.2003.12.004](https://doi.org/10.1016/j.newar.2003.12.004). ARXIV: [astro-ph/0403052](https://arxiv.org/abs/astro-ph/0403052) [astro-ph]
92. D. Hoffleit and W. Warren Jr. *The Bright Star Catalog, 5th Revised Edition*. 1991.
<http://cdsarc.u-strasbg.fr/viz-bin/Cat?V/50>
93. W. Hofmann. *Performance Limits for Cherenkov Instruments*. 2006.
ARXIV: [astro-ph/0603076](https://arxiv.org/abs/astro-ph/0603076) [astro-ph]

Bibliography

94. J. Holder et al. (VERITAS Collaboration). “VERITAS: Status and Highlights”. In: *Proceedings, 32nd International Cosmic Ray Conference*. Beijing, China, 2011.
DOI: [10.7529/ICRC2011/V12/H11](https://doi.org/10.7529/ICRC2011/V12/H11). ARXIV: [1111.1225](https://arxiv.org/abs/1111.1225) [astro-ph.HE].
Note: The DOI for this reference is currently broken
95. J.D. Hunter. “Matplotlib: A 2D graphics environment”. *Computing in Science & Engineering* 9:3, 2007, pages 90–95.
DOI: [10.1109/MCSE.2007.55](https://doi.org/10.1109/MCSE.2007.55)
96. International Earth Rotation and Reference Systems Service. *The ICRF*.
<http://hpiers.obspm.fr/icrs-pc/newwww/icrf/index.php>
97. ITMC. *Linux-HPC-Cluster an der TU Dortmund*. 2018.
<https://www.lido.tu-dortmund.de/cms/de/LiD03/index.html> visited on 2019-11-19
98. E. Jourdain and J. Roques. “The high-energy emission of the Crab Nebula from 20 keV TO 6 MeV with integral SPI”. *The Astrophysical Journal* 704:1, 2009, page 17.
DOI: [10.1088/0004-637X/704/1/17](https://doi.org/10.1088/0004-637X/704/1/17)
99. M. G. Kirsch et al. “Crab: the standard X-ray candle with all (modern) X-ray satellites”. In: *Proceedings, UV, X-Ray, and Gamma-Ray Space Instrumentation for Astronomy XIV*. Vol. 5898. International Society for Optics and Photonics. 2005, page 589803.
DOI: [10.1117/12.616893](https://doi.org/10.1117/12.616893)
100. R. P. Kirshner. “Spectrophotometry of the Crab Nebula”. *The Astrophysical Journal* 194, 1974, pages 323–328.
DOI: [10.1086/153248](https://doi.org/10.1086/153248)
101. M. L. Knoetig et al. (FACT Collaboration). “FACT - Long-term stability and observations during strong Moon light”. In: *Proceedings, 33rd International Cosmic Ray Conference*. 695. Rio de Janeiro, 2013, page 0695.
ARXIV: [1307.6116](https://arxiv.org/abs/1307.6116) [astro-ph.IM].
<http://www.cbpf.br/%7Eicrc2013/papers/icrc2013-0695.pdf>
102. K. Kosack and M. Peresano. “ctapipe: A Low-level Data Processing Framework for CTA”. In: *Proceedings, 36th International Cosmic Ray Conference (ICRC2019)*. Vol. 36. 2019
103. K. Kosack et al. *ctapipe*. Version vo.7.0. 2019.
DOI: [10.5281/zenodo.3372211](https://doi.org/10.5281/zenodo.3372211). <https://github.com/cta-observatory/ctapipe>
104. L. Kuiper et al. “The Crab pulsar in the 0.75-30 MeV range as seen by CGRO COMPTEL-A coherent high-energy picture from soft X-rays up to high-energy γ -rays”. *Astronomy & Astrophysics* 378:3, 2001, pages 918–935.
DOI: [10.1051/0004-6361:20011256](https://doi.org/10.1051/0004-6361:20011256)
105. H. Laurent and R. L. Rivest. “Constructing optimal binary decision trees is NP-complete”. *Information processing letters* 5:1, 1976, pages 15–17
106. C. Leifer et al. *peewee – a small, expressive orm*. Version 3.
<https://github.com/coleifer/peewee>

107. R. W. Lessard et al. “A new analysis method for reconstructing the arrival direction of TeV gamma rays using a single imaging atmospheric Cherenkov telescope”. *Astroparticle Physics* 15:1, 2001, pages 1–18.
DOI: [10.1016/s0927-6505\(00\)00133-x](https://doi.org/10.1016/s0927-6505(00)00133-x)
108. T.-P. Li and Y.-Q. Ma. “Analysis methods for results in gamma-ray astronomy”. *The Astrophysical Journal* 272, 1983, pages 317–324.
DOI: [10.1086/161295](https://doi.org/10.1086/161295)
109. J. F. Macias-Perez et al. “Global Spectral Energy Distribution of the Crab Nebula in the Prospect of the Planck Satellite Polarization Calibration”. *The Astrophysical Journal* 711:1, 2010, page 417
110. G. Maier et al. (CTA Collaboration). “Performance of the Cherenkov Telescope Array”. In: *Proceedings, 35th International Cosmic Ray Conference*. 846. Sissa Medialab, 2017.
DOI: [10.22323/1.301.0846](https://doi.org/10.22323/1.301.0846)
111. D. Maurin, F. Melot, and R. Taillet. “A database of charged cosmic rays”. *Astronomy & Astrophysics* 569:A32, 2014.
DOI: [10.1051/0004-6361/201321344](https://doi.org/10.1051/0004-6361/201321344)
112. R. McGrath et al. *GNU Make*.
<https://www.gnu.org/software/make>
113. W. McKinney. “Data Structures for Statistical Computing in Python”. In: *Proceedings, 9th Python in Science Conference*. Ed. by S. van der Walt and J. Millman. 2010, pages 51–56
114. Meyer, M., Horns, D., and Zechlin, H.-S. “The Crab Nebula as a standard candle in very high-energy astrophysics”. *Astronomy & Astrophysics* 523, 2010, A2.
DOI: [10.1051/0004-6361/201014108](https://doi.org/10.1051/0004-6361/201014108)
115. P. Mezger et al. “Maps of Cassiopeia A and the Crab Nebula at lambda 1.2 mm”. *Astronomy & Astrophysics* 167, 1986, pages 145–150.
<http://articles.adsabs.harvard.edu/pdf/1986A%26A...167..145M>
116. I. Mierswa et al. “Yale: Rapid prototyping for complex data mining tasks”. In: *Proceedings, 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2006, pages 935–940
117. R. Mirzoyan (MAGIC Collaboration). *First time detection of a GRB at sub-TeV energies; MAGIC detects the GRB 190114C*. 2019.
<http://www.astronomerstelegam.org/?read=12390>
118. NASA and Space Telescope Science Institute. *Black Hole-Powered Jet of Electrons and Sub-Atomic Particles Streams From Center of Galaxy M87*. 2000.
<https://hubblesite.org/contents/media/images/2000/20/968-Image.html> visited on 2019-12-11

Bibliography

119. NASA, ESA, G. Dubner et al., A. Loll et al., T. Temim et al., F. Seward et al., VLA/NRAO/AUI/NSF, Chandra/CXC, Spitzer/JPL-Caltech, XMM-Newton/ESA and Hubble/STScI. *Multiwavelength Crab Nebula*. 2017.
<https://hubblesite.org/image/4027/gallery> visited on 2019-11-12
120. D. Nieto et al. “Studying deep convolutional neural networks with hexagonal lattices for imaging atmospheric Cherenkov telescope event reconstruction”. In: *Proceedings, 36th International Cosmic Ray Conference*. Madison, USA, 2019
121. C. Nigro et al. “Towards open and reproducible multi-instrument analysis in gamma-ray astronomy”. *Astronomy & Astrophysics* 625, 2019, A10
122. M. Nöthe. *FACT funfolding*. Version vo.3.4.
https://github.com/fact-project/fact_funfolding
123. M. Nöthe. *mopro3. Large Scale FACT Monte Carlo Production*. Version v3.0.0. 2019.
<https://github.com/fact-project/mopro3>
124. M. Nöthe and K. A. Brügge. *FACT Open Crab Sample Analysis*.
https://github.com/fact-project/open_crab_sample_analysis
125. M. Nöthe, K. A. Brügge, and J. B. Buß. *aict-tools. Reproducible Artificial Intelligence for Cherenkov Telescopes*. Version vo.20.0. 2019.
doi: 10.5281/zenodo.3338081. <https://github.com/fact-project/aict-tools>
126. M. Nöthe, K. A. Brügge, and D. Neise. *custos. A framework to monitor complex systems and send alerts*. Version o.1.1.
<https://github.com/fact-project/pycustos>
127. M. Nöthe et al. *ERNA: Easy RuN Access. A collection of tools to handle FACT data and to execute jobs on SLURM/SGE/TORQUE clusters*. Version o.10.0.
<https://github.com/fact-project/erna>
128. M. Nöthe et al. (FACT Collaboration). “FACT – Performance of the First Cherenkov Telescope Observing with SiPMs”. In: *Proceedings, 35th International Cosmic Ray Conference*. Busan, Republic of Korea, 2017.
doi: 10.22323/1.301.0791
129. *Open Neural Network Exchange Format*.
github.com/onnx/onnx
130. Oracle. *Jar File Specification*.
<https://docs.oracle.com/javase/8/docs/technotes/guides/jar/jar.html> visited on 2020-01-13
131. D. W. Pankenier. “Notes on translations of the East Asian records relating to the supernova of AD 1054”. *Journal of Astronomical History and Heritage* 9:1, 2006, pages 77–82.
<https://ui.adsabs.harvard.edu/abs/2006JAHH...9...77P/abstract>
132. F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. *Journal of Machine Learning Research* 12, 2011, pages 2825–2830

133. T. Pierog et al. “EPOS LHC: Test of collective hadronization with data measured at the CERN Large Hadron Collider”. *Physical Review C* 92:3, 2015.
DOI: [10.1103/physrevc.92.034906](https://doi.org/10.1103/physrevc.92.034906). ARXIV: [1306.0121](https://arxiv.org/abs/1306.0121) [hep-ph]
134. C. Pittori, M. Tavani, et al. (AGILE Collaboration). “Scientific goals and instrument performance of the gamma-ray imaging detector AGILE”. *Nuclear Physics B-Proceedings Supplements* 134, 2004, pages 72–74.
DOI: [10.1016/j.nuclphysbps.2004.08.010](https://doi.org/10.1016/j.nuclphysbps.2004.08.010)
135. *Precision and recall*. 2014.
<https://commons.wikimedia.org/wiki/File:Precisionrecall.svg> visited on 2019-07-11
136. A. M. Price-Whelan et al. (Astropy Collaboration). “The Astropy Project: Building an Open-science Project and Status of the v2.0 Core Package”. *The Astronomical Journal* 156:3, 2018, page 123.
DOI: [10.3847/1538-3881/aabc4f](https://doi.org/10.3847/1538-3881/aabc4f)
137. S. Ritt, R. Dinapoli, and U. Hartmann. “Application of the DRS chip for fast waveform digitizing”. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 623:1, 2010, pages 486–488
138. T. Ruhe et al. “Cherenkov Telescope Ring - An Idea for World Wide Monitoring of the VHE Sky”. In: *Proceedings, Very Large Volume Neutrino Telescopes 2018*. Vol. 207. 2019, page 03002.
DOI: [10.1051/epjconf/201920703002](https://doi.org/10.1051/epjconf/201920703002)
139. T. Y. Saito and J. Sitarek. *Improvement of the θ^2 analysis by using the Random Forest method in the DISP estimation*. Technical report TDAS 09-01. MAGIC, 2009.
<https://magic.mpp.mpg.de/backend/document/file/94b83267ad0bc8d7e59990951a5da57bdd0113f0>.
Note: Internal MAGIC document
140. H. Schoorlemmer and R. White (CTA GCT Collaboration). “A Compact High Energy Camera (CHEC) for the Gamma-ray Cherenkov Telescope of the Cherenkov Telescope Array”. In: *Proceedings of 35th International Cosmic Ray Conference — PoS(ICRC2017)*. 2017.
DOI: [10.22323/1.301.0817](https://doi.org/10.22323/1.301.0817)
141. I. Shilon et al. “Application of deep learning methods to analysis of imaging atmospheric Cherenkov telescopes data”. *Astroparticle Physics* 105, 2019, pages 44–53.
DOI: [10.1016/j.astropartphys.2018.10.003](https://doi.org/10.1016/j.astropartphys.2018.10.003). ARXIV: [1803.10698](https://arxiv.org/abs/1803.10698) [astro-ph.IM]
142. J. Sitarek et al. “Analysis techniques and performance of the Domino Ring Sampler version 4 based readout for the MAGIC telescopes”. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 723, 2013, pages 109–120.
DOI: [10.1016/j.nima.2013.05.014](https://doi.org/10.1016/j.nima.2013.05.014). ARXIV: [1305.1007](https://arxiv.org/abs/1305.1007) [astro-ph.IM]
143. *SLURM Workload Manager*. 2019.
<https://slurm.schedmd.com/overview.html> visited on 2019-07-01

Bibliography

144. D. Sobczyńska. *Mmcs from CORSIKA 6.014*. Technical report TDAS 02-10. MAGIC, 2002.
Note: Internal MAGIC document
145. M. Tanabashi et al. (Particle Data Group Collaboration). “Review of Particle Physics”. *Phys. Rev. D* 98, 3 2018, page 030001.
doi: [10.1103/PhysRevD.98.030001](https://doi.org/10.1103/PhysRevD.98.030001)
146. F. Tavecchio. “Gamma Rays From Blazars”. In: *Proceedings, 6th International Symposium on High-Energy Gamma-Ray Astronomy*. Vol. 1792. 1. 2016, page 020007.
doi: [10.1063/1.4968892](https://doi.org/10.1063/1.4968892)
147. *Telegram Messenger APIs*.
<https://core.telegram.org/> visited on 2019-11-28
148. T. Temim et al. “Spitzer space telescope infrared imaging and spectroscopy of the Crab Nebula”. *The Astronomical Journal* 132:4, 2006, page 1610
149. T. F. Temme. “On the hunt for photons: analysis of Crab Nebula data obtained by the first G-APD Cherenkov telescope”. PhD thesis. TU Dortmund, 2016.
doi: [10.17877/DE290R-17773](https://doi.org/10.17877/DE290R-17773)
150. *Twilio - Communication APIs for SMS, Voice, Video and Authentication*. 2019.
<https://twilio.com> visited on 2019-11-28
151. S. E. Urban and P. K. Seidelmann. *Explanatory supplement to the astronomical almanac*. 3rd ed. University Science Books, 2012
152. G. Varoquaux, O. Grisel, et al. *joblib*.
<https://joblib.readthedocs.io>
153. M. Veron-Cetty and L. Woltjer. “Spectrophotometry of the continuum in the Crab Nebula”. *Astronomy and Astrophysics* 270, 1993, pages 370–378.
<https://ui.adsabs.harvard.edu/abs/1993A%26A...270..370V/abstract>
154. P. Virtanen et al. “SciPy 1.0: fundamental algorithms for scientific computing in Python”. *Nature Methods*, 2020.
doi: [10.1038/s41592-019-0686-2](https://doi.org/10.1038/s41592-019-0686-2)
155. S. P. Wakely and D. Horan. “TeVcat: An online catalog for Very High Energy Gamma-Ray Astronomy”. In: *Proceedings, 30th International Cosmic Ray Conference*. Vol. 3. Mexico City, 2008, pages 1341–1344.
<http://tevcat.uchicago.edu/> visited on 2019-12-06
156. S. v. d. Walt, S. C. Colbert, and G. Varoquaux. “The NumPy array: a structure for efficient numerical computation”. *Computing in Science & Engineering* 13:2, 2011, pages 22–30
157. D. C. Wells and E. W. Greisen. “FITS – a flexible image transport system”. In: *Image Processing in Astronomy*. 1979, page 445

158. R. White (CHEC Collaboration). *CHEC*. 2019.
https://indico.cta-observatory.org/event/1995/contributions/20205/attachments/15670/19999/d2019-06-03_RW_CHECPlenary_Lugano_v04_export.pdf.
Note: Presentation given at the CTA Consortium meeting, internal CTA document.
159. W. Wild. *Small-Sized Telescope Harmonization Process and Status*. 2019.
<https://www.cta-observatory.org/small-sized-telescope-harmonization-process-and-status> visited
on 2020-01-18

Glossary

- ADC** analog-to-digital converter. 20, 32
- AGN** Active Galactic Nucleus. 8, 10, 12, 14, 39
- ceres** Camera Electronics and Reflector Simulation. 26–28, 40, 59, 117
- coconut** CORSIKA configuration utility. 29
- CORSIKA** Cosmic Ray Simulations for Kascade. 23–29, 39, 40, 55, 56, 73, 93, 94, 113, 116, 117
- CTA** Cherenkov Telescope Array. 8, 14, 17, 45, 69, 76, 94
- DIM** Distributed Information Management System. 87, 90
- DRS** Domino Ring Sampler. 38, 42
- DRS4** Domino Ring Sampler 4. 20, 31–33
- EAS** Extensive Air Shower. 5, 7, 23
- EBL** extragalactic background light. 14
- egs4** Electron Gamma Shower 4. 24
- FACT** First G-APD Cherenkov Telescope. 3, 8, 9, 12, 16–21, 23–29, 31–33, 39, 42, 43, 45, 51, 55, 57, 59, 63–65, 69, 77, 81, 83, 87–89, 91, 93, 94, 117
- FADC** fast analog-to-digital converter. 19, 20
- FITS** Flexible Image Transport System. 26, 31, 38, 43, 45, 57
- FLUKA** Fluctuating Kascade. 24, 29, 30, 93
- FOSS** Free and Open Source Software. 94
- FSRQ** Flat Spectrum Radio Quasar. 10
- FTM** FACT trigger master. 19, 20
- G-APD** Geiger-Mode Avalanche Photo Diode. 18
- GHEISHA** Gamma Hadron Electron Interaction Shower. 24
- GRB** gamma-ray burst. 6, 8, 14
- HAWC** High Altitude Water Cherenkov. 9
- HEGRA** High Energy Gamma-Ray Astronomy. 18, 24

Glossary

- H. E. S. S.** High Energy Stereoscopic System. 7
- IACT** Imaging Air Cherenkov Telescope. 4, 5, 7–9, 14, 16, 18, 23, 24, 38, 39, 45, 47, 49, 53, 69, 76, 77
- ICRS** International Celestial Reference System. 38, 40, 53, 63
- IRF** instrument response function. 69, 84
- ISDC** Integral Science Data Center. 42
- JSON** JavaScript Object Notation. 38
- LAT** Large Area Telescope. 5
- LHC** Large Hadron Collider. 24
- LIGO** LASER Interferometer Gravitational-Wave Observatory. 4
- MAGIC** Major Atmospheric Gamma-Ray Imaging Cherenkov. 7, 8, 11, 12, 16–18, 26, 52, 57, 79, 82, 84, 89
- MARS** Magic Analysis and Reconstruction Software. 26, 27, 65, 67
- MCMC** Markov chain Monte Carlo. 83
- MMCS** MAGIC Monte Carlo Software. 24, 25, 28, 29, 56
- MSE** mean squared error. 51
- NSB** Night Sky Background. 26, 36, 56, 118
- OGA** open gamma-ray astronomy. 53
- ONNX** Open Neural Network Exchange Format. 49
- ORM** Observatorio del Roque de los Muchachos. 7, 8
- PDF** probability density function. 116
- p. e.** photon equivalents. 33
- PMML** Predictive Model Markup Language. 49
- PMT** photo multiplier tube. 9, 18, 19, 25
- PSF** point spread function. 69, 117
- PWN** Pulsar Wind Nebula. 10–12
- QGSJET** Quark Gluon String model with Jets. 24
- ROC** Receiver Operating Characteristic. 48, 49, 66, 68
- SIPM** Silicon Photo Multiplier. 17–20, 26, 56, 89, 117, 118

- SNR** supernova remnant. 9, 11, 12
- SQL** Structured Query Language. 57
- SST** Small Sized Telescope. 17
- TAI** Temps Atomique International. 40
- URQMD** Ultra-relativistic Quantum Molecular Dynamics. 24, 29, 30, 93
- UT1** Universal Time 1. 40
- UTC** Coordinated Universal Time¹. 40
- VERITAS** Very Energetic Radiation Imaging Telescope Array System. 8, 18
- VLBI** Very Long Baseline Interferometry. 10, 39, 40
- WIMP** weakly interacting massive particle. 14

¹This abbreviation was chosen as a compromise between French (temps universel coordonné) and English that would not favor one language.

Appendix

1 On Reproducing this Work

This document is built completely without human intervention from only a few input data files, scripts to create visualizations and the \LaTeX source code. The input data files comprise the image parameters for the different datasets as described in the thesis and some additional files, e. g. the CORSIKA output for the Cherenkov footprint visualizations.

To build this thesis, including training all the machine learning models, applying them to the data, calculate and visualize the performance metrics, unfold the spectrum of the Crab Nebula and compile the \LaTeX document, only a single call to the make program is necessary.

1.1 Source Code of the Thesis

The source code of this thesis is publicly available at https://github.com/maxnoe/phd_thesis. Download an archive from the website or use git:

```
$ git clone --recursive https://github.com/maxnoe/phd_thesis
```

1.2 Access to the Data

The input data files needed to build this thesis are at the time of writing archived on the storage server `big-tank.app.tu-dortmund.de` in the directory

```
/POOL/users/mnoethe/phd_thesis/data.
```

This directory is required to be copied or linked into the data directory in the base directory of this thesis.

1.3 Used Software

GNU Make [112] is used to define the steps needed to produce this thesis. The analysis itself builds on the scientific Python stack [156, 154, 113, 95, 132], astropy [136] and a number of packages developed and contributed to by me during the work on this thesis.

Building the thesis needs TeXLive 2020, from <https://www.tug.org/texlive>, for installation instructions see <https://toolbox.pep-dortmund.org/install/linux>.

Python in version 3.7 via the conda package manager is used. On Linux, to download and install miniconda and setup the environment needed for this thesis, use:

```
$ curl -LO https://repo.anaconda.com/miniconda/Miniconda3-4.7.12-Linux-x86_64.sh
$ bash Miniconda3-4.7.12-Linux-x86_64.sh -b -p path/to/install/miniconda
$ source path/to/install/miniconda/etc/profile.d/conda.sh
$ conda env create -n phd_mnoethe -f full_environment.yaml
```

Appendix

The `full_environment.yaml` file lists all packages in the exact version as was used to produce this thesis. The file `environment.yaml` lists the minimal, direct dependencies with more lax version requirements, these are:

```
name: phd_mnoethe
channels:
  - defaults
dependencies:
  - python=3.7
  - astropy=4
  - h5py=2.9
  - hdf5=1.10
  - ipython=7.8
  - joblib=0.13.2
  - matplotlib=3.1
  - notebook=6.0.1
  - numpy=1.17
  - pandas=0.25
  - pandoc=2.2.3
  - pip>=19
  - qt=5.9
  - scikit-learn=0.21
  - scipy=1.3
  - setuptools
  - sqlalchemy=1.3
  - sqlite=3
  - wheel=0.33
  - zstd=1.3
  - tqdm=4
  - pytables=3.5
  - shapely=1.6
  - numba
  - pip:
    - aict-tools==0.21.0
    - ctapipe==0.8.0
    - pyfact==0.25.2
    - eventio==1.0.0
    - zstandard
    - https://github.com/cta-observatory/ctapipe-extra/archive/v0.3.0.tar.gz
    - https://github.com/fact-project/fact_plots/archive/v0.12.2.tar.gz
    - https://github.com/fact-project/irf/archive/v0.5.3.tar.gz
    - https://github.com/fact-project/fact_funfolding/archive/v0.3.6.tar.gz
```

1.4 Building this thesis

After installing all required software and acquiring the input data, the only thing needed to build this document and all its figures and tables is to call

```
$ conda activate phd_mnoethe
```

```
$ make
```

in the base directory of the git repository. It will take a few hours, to train all the machine learning models, apply them, create figures and tables and to typeset the final document.

1.5 Docker Container

A docker image with all installed software to directly build this thesis is defined in the `Dockerfile` in the git repository and is stored on the big-tank server.

2 Cherenkov Footprint of 10 TeV Iron Shower by Emitting Particle

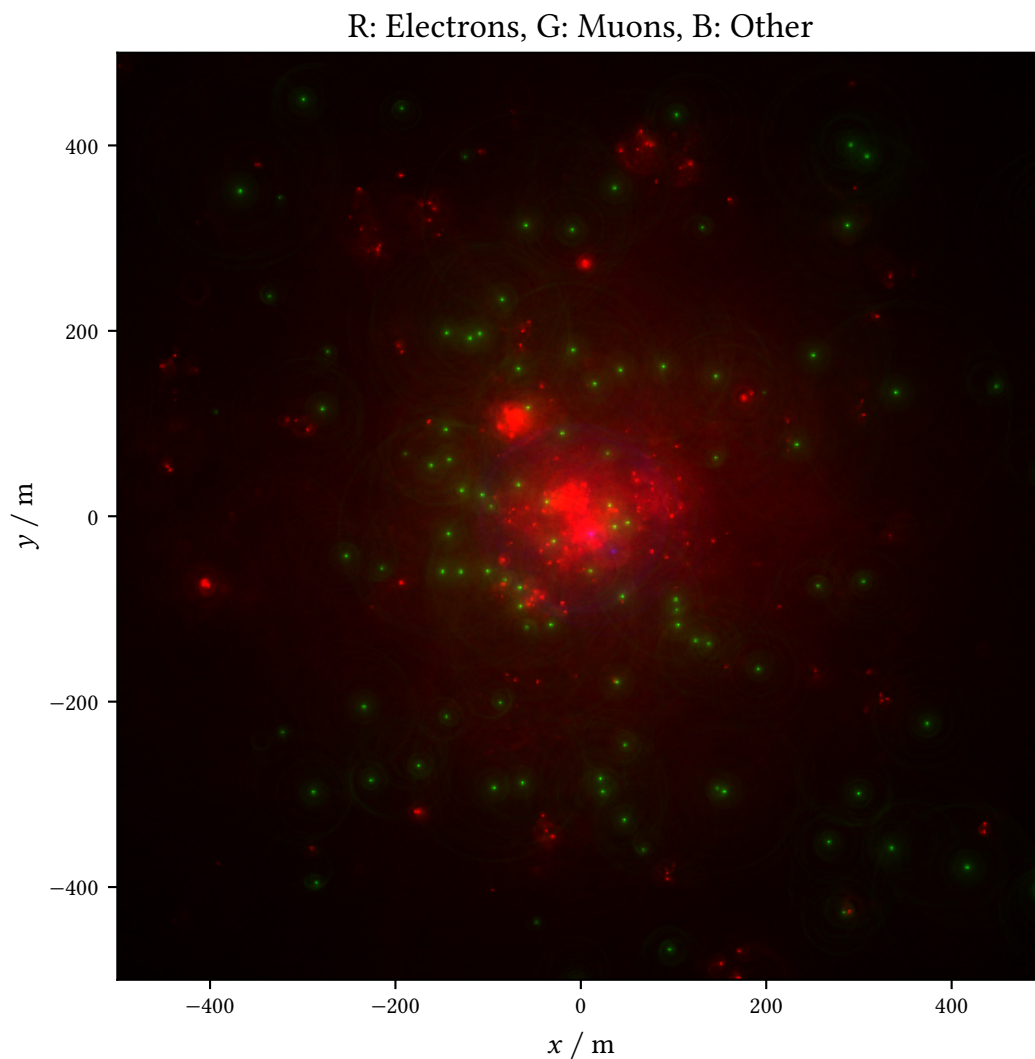


Figure 1: Cherenkov light at the observation level for a 10 TeV iron nucleus. The color channels are for the different emitting particles: red is light caused by electrons and positrons, green for muons and blue for all others, mainly protons and pions.

3 Covariance Matrices of the Cosmic Ray Flux Fits

For the proton flux:

$$\text{Cov}(\gamma, \log_{10}(\Phi_0)) = \begin{pmatrix} 0.000051 & -0.000173 \\ -0.000173 & 0.000647 \end{pmatrix} \quad (1)$$

For the helium flux:

$$\text{Cov}(\gamma, \log_{10}(\Phi_0)) = \begin{pmatrix} 0.000127 & -0.000412 \\ -0.000412 & 0.001470 \end{pmatrix} \quad (2)$$

4 Probability Density Functions for Primary Properties in CORSIKA

The energy is sampled from a power law distribution with the following probability density function (PDF):

$$P_E(E) = \begin{cases} \frac{(\gamma+1)E_{\text{ref}}^\gamma}{E_{\text{max}}^{\gamma+1} - E_{\text{min}}^{\gamma+1}} \left(\frac{E}{E_{\text{ref}}}\right)^\gamma & E_{\text{min}} \leq E \leq E_{\text{max}} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

With $E_{\text{ref}} = 1$ GeV for CORSIKA. The impact distance is sampled such that it is uniform in area, resulting in the following PDF:

$$P_R(R) = \begin{cases} \frac{2R}{R_{\text{max}}^2} & 0 \leq R \leq R_{\text{max}} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

The azimuth angle is sampled uniformly:

$$P_\varphi(\varphi) = \begin{cases} \frac{1}{\varphi_{\text{max}} - \varphi_{\text{min}}} & \varphi_{\text{min}} \leq \varphi \leq \varphi_{\text{max}} \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

The zenith distance is sampled such that events are uniform in solid angle in the given limits:

$$P_\vartheta(\vartheta) = \begin{cases} \frac{\sin \vartheta}{\cos \vartheta_{\text{min}} - \cos \vartheta_{\text{max}}} & \vartheta_{\text{min}} \leq \vartheta \leq \vartheta_{\text{max}} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

This results in a sawtooth-like distribution, when several zenith intervals are simulated.

5 Database Tables of Mopro3

5.1 CorsikaSettings

id, Integer, PrimaryKey

name, Text Human readable description, used for directory structure and output files.

version, Integer CORSIKA version identifier. The current release is 77000, used for production was 76900.

config_h, Text The CORSIKA compilation time configuration as precompiler header, this is generated by running the CORSIKA configuration utility (coconut). The compile time configuration of CORSIKA determines among other options the used interaction models. The main purpose of storing the whole header file and not single options is to minimize the complexity and maintainability of mopro3 by not implementing each possible option but rely on the header file generation through the coconut tool.

inputcard_template, Text The template for the input card as discussed in [section 3.4](#)

additional_files, Binary On its first run, CORSIKA calculates a number of interpolation tables, which is quite time intensive and can take up to several hours. This is infeasible for a large, parallel production on a cluster. Thus, these interpolation tables are calculated once up front and stored as a tarball in the database, which is unpacked into the CORSIKA run directory for each job.

5.2 CorsikaRun

id, Integer, PrimaryKey inserted as the run number into the input card template

corsika_settings_id, ForeignKey reference to the CorsikaSettings to be used for this run.

primary_particle, Integer CORSIKA identification number of the primary particle, e. g. 1 for gamma ray, 14 for proton.

n_showers, Integer how many showers to generate in this run

zenith_min, Float

zenith_max, Float minimum and maximum zenith range

azimuth_min, Float

azimuth_max, Float minimum and maximum azimuth range

energy_min, Float

energy_max, Float minimum and maximum energy range

reuse, Integer how many times the shower should be reused, see [subsection 3.1.2](#)

max_radius, Float maximum scatter radius of the primary particle

bunch_size Bunch size option as discussed in [subsection 3.1.2](#)

The ceres configuration is stored in the table CeresSettings and contains these columns:

id, Integer, PrimaryKey

name, Text human readable name, used for the directory structure and output files

revision, Integer version number in FACT's version control repository

rc_template, Text template for the ceres rc file

resource_files, Binary A tarball with further configuration files. This includes for example tables of the reflectivity versus wavelength of the mirrors and the photon detection efficiency of the SIPMs.

psf_sigma, Float PSF of the individual mirror facets

apd_dead_time, Float

apd_recovery_time, Float

apd_cross_talk, Float

apd_afterpulse_probability_1, Float

- apd_afterpulse_probability_2, Float** Physical properties of the SIPM cells, see [chapter 2](#).
- excess_noise, Float** Standard deviation of white noise added to the electronics signal.
- nsb_rate, Float** Rate of NSB photons
- additional_photon_acceptance, Float** Photons are only processed with this probability, effectively discarding a fixed percentage of photons. Introduced for [149], this was motivated by the missing obstruction by the telescope structure and degrading mirrors.
- dark_count_rate, Float** of the SIPM
- pulse_shape_function, Text** a function describing the shape of the single photon pulse
- residual_time_spread, Float** Standard deviation of a fixed time offset between pixels
- gapd_time_jitter, Float** Standard deviation for a normal random number added to the arrival time of each individual Cherenkov photon
- discriminator_threshold** Trigger threshold, see [section 2.1](#)

5.3 CeresRun

- id, Integer, PrimaryKey**
- corsika_run_id, ForeignKey** id of the CorsikaRun used as input
- ceres_settings_id, ForeignKey** id of the CeresSettings to be used for this run
- diffuse, Boolean** if this run should simulate point-like or diffuse source of the primary. For the diffuse case, pointing direction of the telescope is drawn randomly around the direction of the primary particle.
- off_target_distance, Float** wobble distance for point source mode or maximum angle to the optical axis for diffuse mode.

5.4 Processing related fields for CorsikaRun and CeresRun

Additionally, both CorsikaRun and CeresRun have these columns for tracking job state

- location, Text** Processing location, for the case of multiple consumers from the same database. Only CeresRuns are started where the corresponding CorsikaRun was processed at the same location, so that the input file is available.
- walltime, Integer** walltime is the maximum time a job is allowed to take and is required by most cluster job management tools like SLURM so that jobs can be scheduled.
- status_id, ForeignKey** id of the current job status
- priority, Integer** jobs are processed in order of priority, id. Lower values mean more important.
- duration, Integer** The runtime of the job in seconds, filled after a run finished successfully

Also the path of the outputfiles, one for CorsikaRun, two for each CeresRun are stored.

6 CORSIKA Input Card Template

```
RUNNR    {{ run.id }}
EVTNR    1
```



```

NSHOW    {{ run.n_showers }}
PRMPAR   {{ run.primary_particle }}

ESLOPE   {{ run.spectral_index }}
ERANGE   {{ run.energy_min }} {{ run.energy_max }}

THETAP   {{ run.zenith_min }} {{ run.zenith_max }}
PHIP     {{ run.azimuth_min }} {{ run.azimuth_max }}
VIEWCONE 0.0 {{ run.viewcone }}

TELESCOPE 0. 0. 0. 5e2 0          Single Telescope with 5m radius

SEED {{ run.id * 10 + 1 }} 0 0
SEED {{ run.id * 10 + 2 }} 0 0
SEED {{ run.id * 10 + 3 }} 0 0
SEED {{ run.id * 10 + 4 }} 0 0

TSTART T
CWAVLG 290 900
CERSIZ {{ run.bunch_size }}
CSCAT {{ run.reuse }} {{ run.max_radius * 100 }} 0
ECUTS 0.3 0.3 0.02 0.02

OBSLEV 2200e2
MAGNET 30.419 23.856          Geomag EMM2017 for 28.761647° N, -17.891116° W,
                              2200m height, 2013-12-31

ARRANG -6.15
ATMOSPHERE 8 N                La Palma Winter

DIRECT ./
DATBAS F

PAROUT F F
CERFIL 0                      Disable corsika raw cherenkov output
MAXPRT 0
TELFIL {{ output_file }}
EXIT

```

7 CERES rc template

```

#                               General
# -----
# Use this if you want to setup the logging stream for the jobs
# (overwrites command line options)
# -----
MLog.VerbosityLevel: 4
MLog.DebugLevel:      0
MLog.NoColors:        no

```

Appendix

```
#                                     Ceres
# -----
# Configure Eventloop
# -----
Overwrite: yes

# -----
# Use this to setup binnings. For more details see: MBinning::ReadEnv
# -----
BinningImpact.Raw:  40 0 1000
BinningTrigPos.Raw: 300 -25 275

# -----
# Initialize random number generator (see MJob::InitRandomNumberGenerator)
# -----
RandomNumberGenerator: TRandom3
RandomNumberSeedValue: 1

# -----
# Some setup for the atmosphere. The default should be well suited.
# -----
MSimAtmosphere.FileAerosols: {{ resource_directory }}/atmosphere-aerosols.txt
MSimAtmosphere.FileOzone:   {{ resource_directory }}/atmosphere-ozone.txt

# -----
# Here you can control the pointing of the telescope. To switch on
# off-target observations set a value for the distance !=0 [deg].
# For details see MSimPointingPos
# -----
{% if run.diffuse %}
MSimPointingPos.OffTargetDistance: -{{ run.off_target_distance }}
{% else %}
MSimPointingPos.OffTargetDistance: {{ run.off_target_distance }}
{% endif %}

# -----
# Setup the reflector and camera geometry
# -----
Reflector.Constructor: MReflector

# For the file definition see MReflector::ReadFile

Reflector.FileName: {{ resource_directory }}/reflector.txt
Reflector.SetSigmaPSF: {{ settings.psf_sigma }}
MGeomCam.Constructor: MGeomCamFACT();

#MSimBundlePhotons.FileName: resmc/fact/dwarf-fact.txt

# Set the APD type (1: 30x30 <default>, 2: 60x60, 3:60x60(ct=15%))
```

```

MSimAPD.Type:                0
MSimAPD.NumCells:            60
MSimAPD.DeadTime:            {{ settings.apd_dead_time }}
MSimAPD.RecoveryTime:        {{ settings.apd_recovery_time }}
MSimAPD.CrosstalkCoefficient: {{ settings.apd_cross_talk }}
MSimAPD.AfterpulseProb1:     {{ settings.apd_afterpulse_probability_1 }}
MSimAPD.AfterpulseProb2:     {{ settings.apd_afterpulse_probability_2 }}

MSimExcessNoise.ExcessNoise:  {{ settings.excess_noise }}

# -----
# Setup the absorption, conversion efficiency and angular acceptance
# -----
MirrorReflectivity.FileName:  {{ resource_directory }}/mirror-reflectivity.txt
PhotonDetectionEfficiency.FileName:  {{ resource_directory }}/pde.txt
ConesAngularAcceptance.FileName:  {{ resource_directory }}/cone-angular-acceptance.txt
ConesTransmission.FileName:  {{ resource_directory }}/cone-transmission.txt

AdditionalPhotonAcceptance.Function.Name:  {{ settings.additional_photon_acceptance }}
AdditionalPhotonAcceptance.Function.Npx:  100
AdditionalPhotonAcceptance.Function.Xmin:  290
AdditionalPhotonAcceptance.Function.Xmax:  900

# -----
# Setup what in MMCS would be called "additional spot size"
# -----
#MSimPSF.Sigma: -1
#MSimReflector.DetectorMargin: 0

# -----
# Setup the dark counts (FrequencyFixed) and the NSB noise per cm^2
# -----
MSimRandomPhotons.FrequencyFixed:  {{ settings.dark_count_rate }}
{% if settings.nsb_rate %}
MSimRandomPhotons.FrequencyNSB:  {{ run.nsb_rate }}
{% else %}
MSimRandomPhotons.FileNameNSB:  {{ resource_directory }}/nsb.txt
{% endif %}

# -----
# Setup the trigger
# -----
# This line could be omitted but then the discriminator would be
# evaluated for all pixels not just for the pixels which are
# later "connected" in the trigger (used in the coincidence map)
MSimTrigger.FileNameRouteAC:  {{ resource_directory }}/route-ac.txt
MSimTrigger.DiscriminatorThreshold:  {{ settings.discriminator_threshold }}

MSimTrigger.CableDelay:  21.0
MSimTrigger.CableDamping:  -0.96
MSimTrigger.CoincidenceTime:  0.5

# Every Pixel(!) should see the same signal independant of its size

```

Appendix

```
MSimCalibrationSignal.NumPhotons: 24
MSimCalibrationSignal.NumEvents: 1000

IntendedPulsePos.Val: 26

# -----
# Setup the FADC
# -----

MRawRunHeader.SamplingFrequency: 2000
MRawRunHeader.NumSamples: 300
MRawRunHeader.NumBytesPerSample: 2
MRawRunHeader.FadcResolution: 12

MSimCamera.DefaultOffset: -1850.0
MSimCamera.DefaultNoise: 2.8125
MSimCamera.DefaultGain: 22.553

# Value for the fudgefactor in the calculation of the accoupling:
MSimCamera.ACFudgeFactor: 0.3136
MSimCamera.ACTimeConstant: 20

# The number of sampling points is almost irrelevant because they
# are equidistant, i.e. calculated and no search is necessary.
# Nevertheless, you must make sure that there are enough points
# to sample the function accurately enough.
# Attention: x in the function is given in slices, so if you change the sampling
# frequency you have to change also this function
PulseShape.Function.Name: {{ settings.pulse_shape_function }}
PulseShape.Function.Npx: 310
PulseShape.Function.Xmin: -10
PulseShape.Function.Xmax: 300

# -----
# This is a cut executed after the calculation of the image parameters
# -----

Cut.Inverted: yes
Cut.Condition: MHillas.fSize>10.0

# Does not trigger anyway
ContEmpty3.Condition: MPhotonEvent.GetNumPhotons<10

MFixTimeOffset.FileName: {{ resource_directory }}/pixel-delays.csv
ResidualTimeSpread.Val: {{ settings.residual_time_spread }}
GapdTimeJitter.Val: {{ settings.gapd_time_jitter }}

# last line comment
```

8 Pedestal Variance

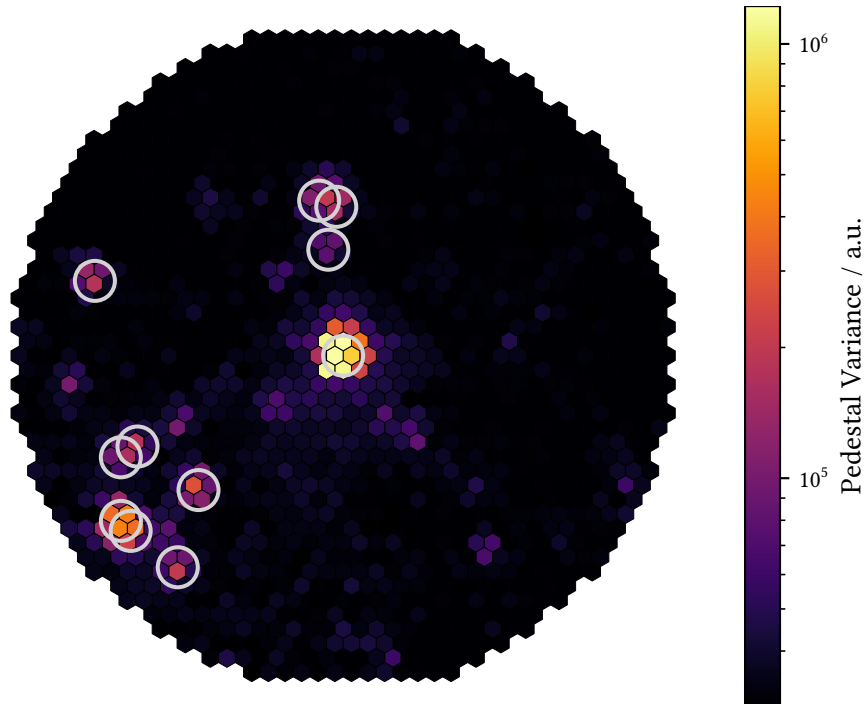


Figure 2: Mean pedestal variance over 1000 events of a run observing Aldebaran, a variable star with an apparent magnitude of around 0.8 mag. To calculate pedestal variance, the time series in each pixel are divided into windows of 30 values each. The values are summed in each window and the variance over all windows is then proportional to the amount of noise photons the pixel received. The gray circles show the positions of stars brighter than 6 mag as queried through the star service. This also demonstrates nicely, that the conversion from celestial coordinates into camera coordinates works correctly, see [section 4.8](#).

9 FACT-Tools Standard Analysis XML

```

<!-- FACT-Tools XML for the erna automatic processing with v1.1.2 -->
<container>
  <properties url="classpath:/default/settings.properties" />

  <property name="infile" value="" />
  <property name="drsfile" value="" />
  <property name="aux_dir" value="" />

  <property
    name="pixelDelayFile"
    value="classpath:/default/delays_lightpulser_20150217.csv"
  />

```

Appendix

```
<property name="output_basename" value="output" />

<service id="auxService"
  class="fact.auxservice.AuxFileService"
  auxFolder="${aux_dir}"
  />
<service id="gainService" class="fact.gainservice.GainService" />
<service id="calibService"
  class="fact.calibrationservice.ConstantCalibService"
  />

<stream id="fact" class="fact.io.hdureader.FITSSStream" url="${infile}"/>

<process id="2" input="fact">
  <include url="classpath:/analysis/init.xml" />

  <!-- Skip events created by malfunctioning LED light pulser (27s Events) -->
  <Skip condition="%{data.saturated_slices_percentage} > 0.2" />
  <!-- Only analyze physics trigger events -->
  <Skip condition="%{data.TriggerType} != 4" />

  <include url="classpath:/analysis/calibration.xml" />
  <include url="classpath:/analysis/extraction.xml" />
  <include url="classpath:/analysis/cleaning.xml" />

  <!-- Only analyze events with at least 5 pixels after cleaning -->
  <Skip condition="%{data.num_pixel_in_shower} < 5" />

  <include url="classpath:/analysis/pedestalParameters.xml" />
  <include url="classpath:/analysis/imageParameters.xml" />
  <include url="classpath:/analysis/sourcePosition.xml" />

  <fact.utils.SanitizeKeys />

  <!-- Set some FITS header keys -->
  <SetValue key="NIGHT" value="%{data.night}" />
  <SetValue key="RUNID" value="%{data.run_id}" />
  <SetValue key="DRSFILE" value="%{data.drs_file}" />
  <SetValue key="OBJECT" value="%{data.source_name}" />
  <SetValue key="RA_OBJ" value="%{data.source_position_ra}" />
  <SetValue key="DEC_OBJ" value="%{data.source_position_dec}" />

  <fact.io.FITSWriter
    headerKeys="NIGHT,RUNID,DRSFILE,OBJECT,RA_OBJ,DEC_OBJ"
    url="file:facttools_output/${output_basename}_level2.fits"
  />
</process>
</container>
```

10 SQL Query for the Crab Dataset

```

SELECT
    fNight,
    fRunID,
    fThresholdMinSet,
    (TIMESTAMPDIFF(SECOND, fRunStart, fRunStop) * fEffectiveOn) AS ontime
FROM RunInfo
WHERE
    fSourceKEY = 5
    AND fRunTypeKey = 1
    AND fNight BETWEEN 20131001 AND 20140205
    AND fZenithDistanceMax < 30
    AND fMoonZenithDistance > 100
    AND fCurrentsMedMeanBeg < 8
    AND fTriggerRateMedian > 40
    AND fTriggerRateMedian < 85
    AND fThresholdMinSet < 350
    AND fEffectiveOn > 0.95
;

```

11 Configuration for the aict-tools

```

seed: 0

n_cross_validations: 10

energy:
  regressor: |
    ensemble.RandomForestRegressor(
      n_estimators=200,
      max_features=5,
      n_jobs=-1,
      criterion='mse',
      max_depth=15,
      random_state=0,
      min_samples_leaf=10,
    )

n_signal: 200000

log_target: True
target_name: corsika_event_header_total_energy

features:
  - size
  - width

```

Appendix

```
- length
- skewness_trans
- skewness_long
- concentration_cog
- concentration_core
- concentration_one_pixel
- concentration_two_pixel
- leakage1
- leakage2
- num_islands
- num_pixel_in_shower
- photoncharge_shower_mean
- photoncharge_shower_variance
- photoncharge_shower_max

# feature generation, constants have to be prefixed with @
feature_generation:
  needed_columns:
    - size
    - width
    - length
    - cog_x
    - cog_y
  features:
    log_size: log(size)
    size_area: size / (width * length * @pi)
    area: (width * length * @pi)
    cog_r: sqrt(cog_x**2 + cog_y**2)

separator:
  # the classifier to use
  classifier : |
    ensemble.RandomForestClassifier(
      n_estimators=200,
      max_features='sqrt',
      n_jobs=-1,
      max_depth=15,
      criterion='entropy',
      min_samples_leaf=10,
    )

# randomly sample the data if you dont want to use the whole set
n_background: 500000
n_signal: 500000
```



```

features:
  - concentration_cog
  - concentration_core
  - concentration_one_pixel
  - concentration_two_pixel
  - leakage1
  - leakage2
  - width
  - length
  - skewness_long
  - skewness_trans
  - kurtosis_long
  - kurtosis_trans
  - num_islands
  - num_pixel_in_shower
  - photoncharge_shower_variance

# feature generation, constants have to be prefixed with @
feature_generation:
  needed_columns:
    - width
    - length
    - size
  features:
    area: width * length * @pi
    log_size: log(size)
    size_area: size / (width * length * @pi)
    area_size_cut_var: (width * length * @pi) / log(size)**2

disp:
  disp_regressor : |
    ensemble.RandomForestRegressor(
      n_estimators=200,
      max_features='sqrt',
      n_jobs=-1,
      max_depth=20,
      min_samples_leaf=10,
    )

  sign_classifier: |
    ensemble.RandomForestClassifier(
      n_estimators=200,
      max_features='sqrt',
      n_jobs=-1,
      max_depth=25,
      min_samples_leaf=10,
    )

```

Appendix

```
source_az_column: source_position_az
source_zd_column: source_position_zd
pointing_az_column: pointing_position_az
pointing_zd_column: pointing_position_zd

# randomly sample the data if you dont want to use the whole set
n_signal : 500000

features:
- num_pixel_in_shower
- width
- length
- skewness_long
- kurtosis_long
- concentration_cog
- concentration_core
- leakage1
- leakage2
- slope_long
- time_gradient_slope_long
- photoncharge_shower_mean
- photoncharge_shower_variance

feature_generation:
  needed_columns:
    - width
    - length
    - size
  features:
    area: width * length * @pi
    width_length: 1 - (width / length)
    log_size: log(size)
    log_size_area: log(size) / (width * length * @pi)
```

12 size for Excess Events and Gamma Simulations

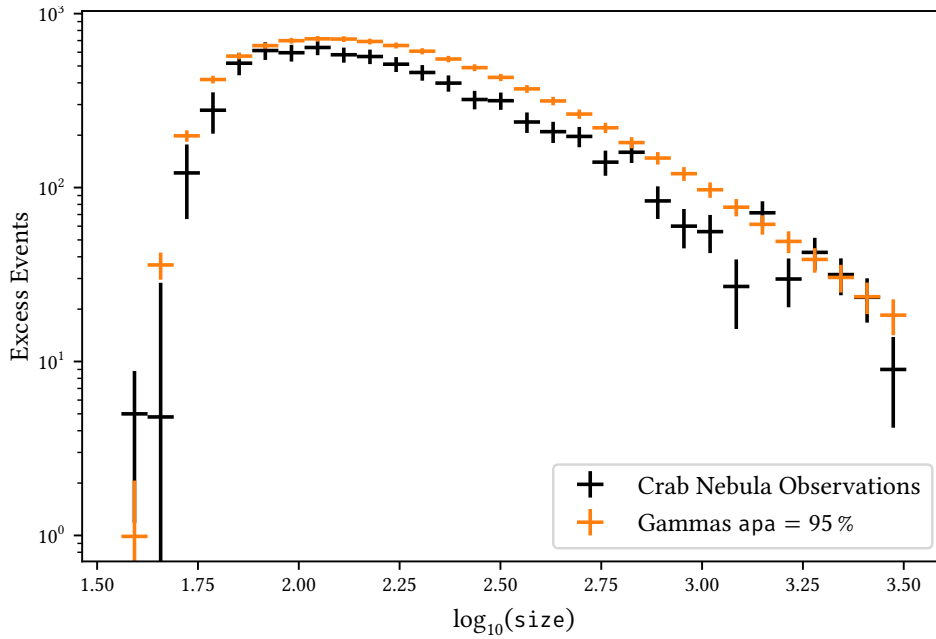


Figure 3: Image parameter size for simulated gammas and the observed excess events in the on region for loose event selection criteria. The simulated dataset with APA = 95 % is shown here.

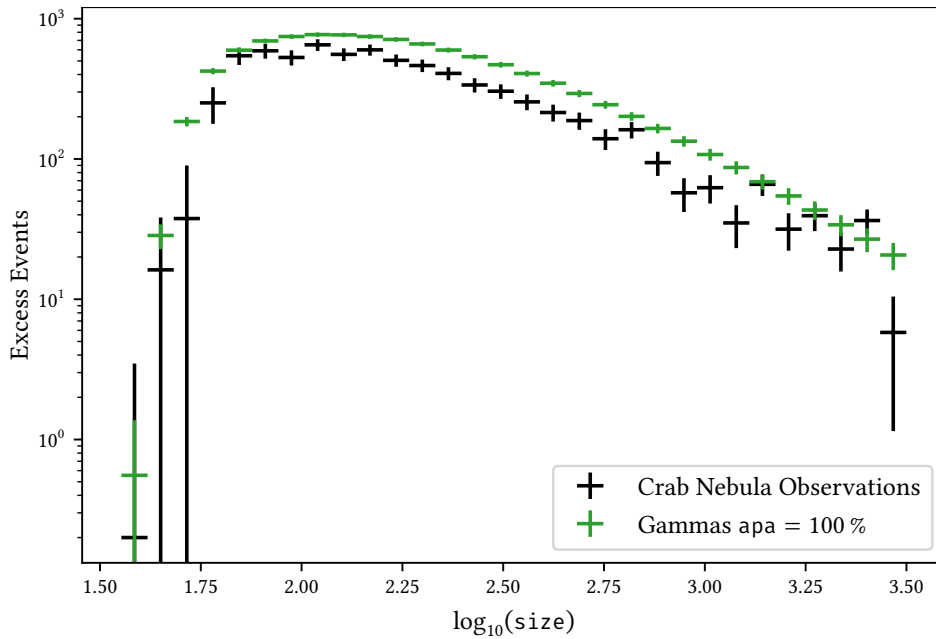


Figure 4: Image parameter size for simulated gammas and the observed excess events in the on region for loose event selection criteria. The simulated dataset with APA = 100 % is shown here.

13 Angular Resolution for all Events

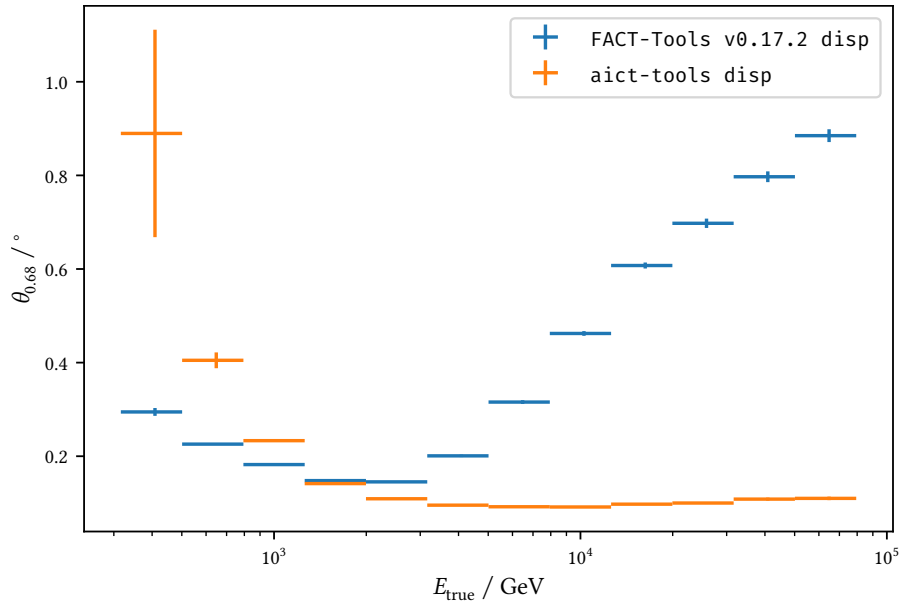


Figure 5: Comparison of the angular resolution with the simple disp parameterization from FACT-Tools with the new, machine learning based parameterization implemented in the aict-tools for the simulated point source gammas of the dataset with APA = 95 %.

14 Bias and Resolution without Applying the Event Selection

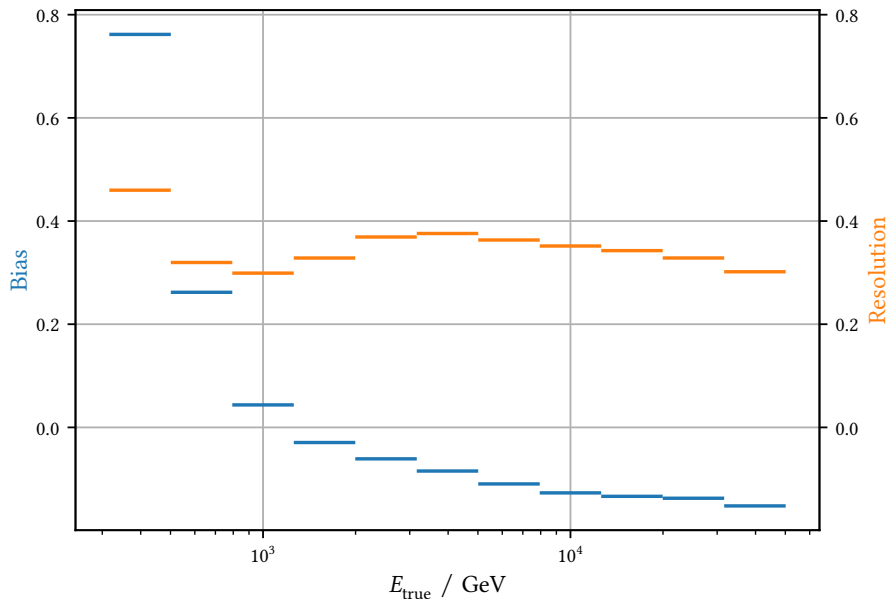


Figure 6: Bias and resolution for all gamma events simulated in wobble mode after applying only the event pre-selection.

15 Performance visualisations

All visualizations presented in [chapter 6](#) for the datasets that were not shown there.

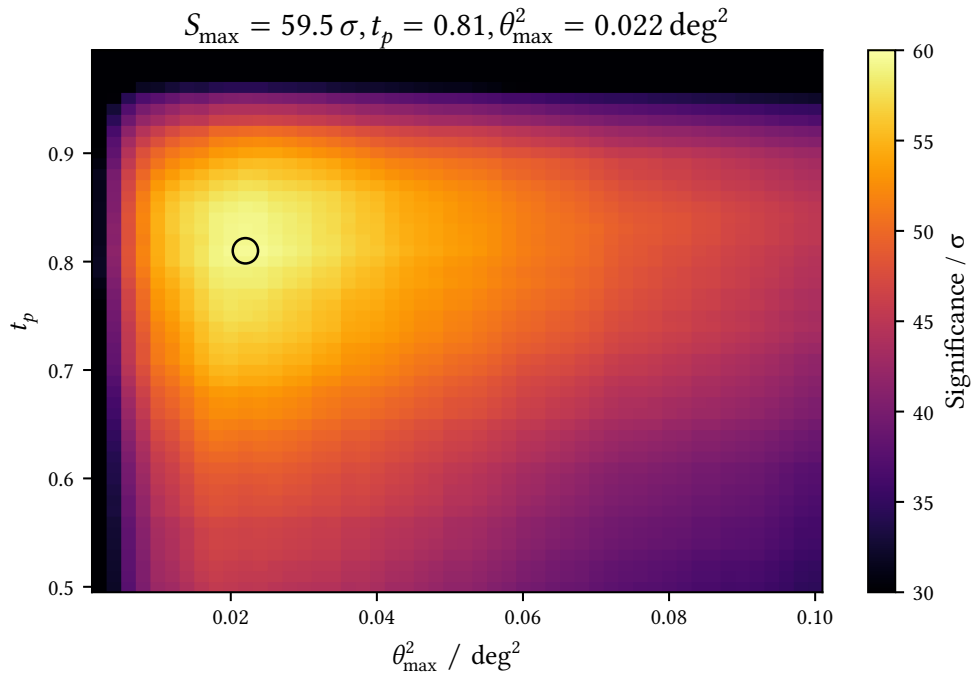


Figure 7: Significances for APA = 95 %, see [Figure 6.10](#).

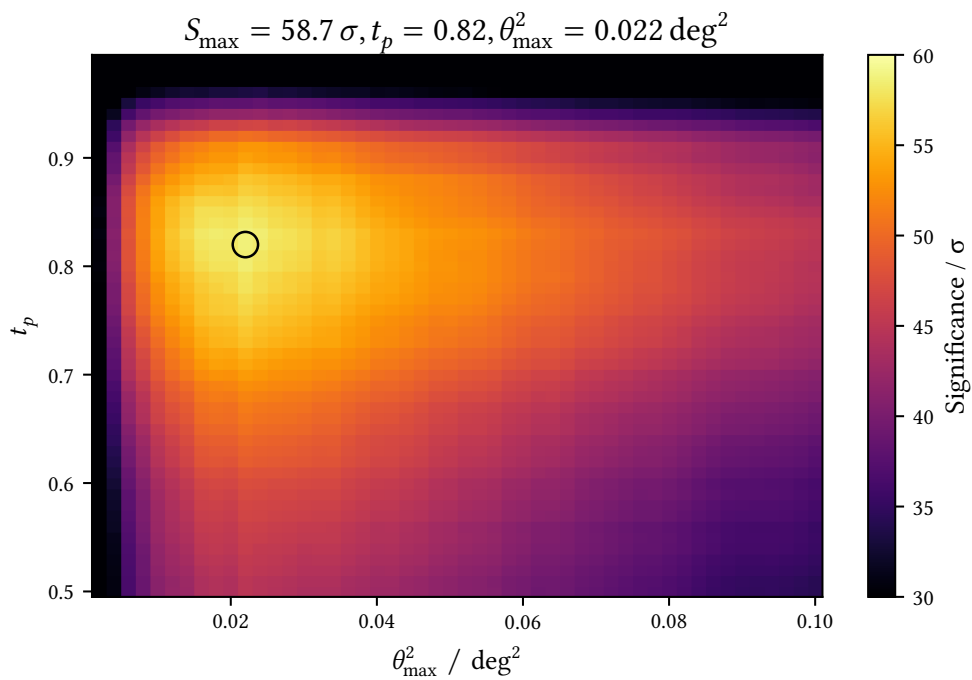


Figure 8: Significances for APA = 100 %, see [Figure 6.10](#).

Appendix

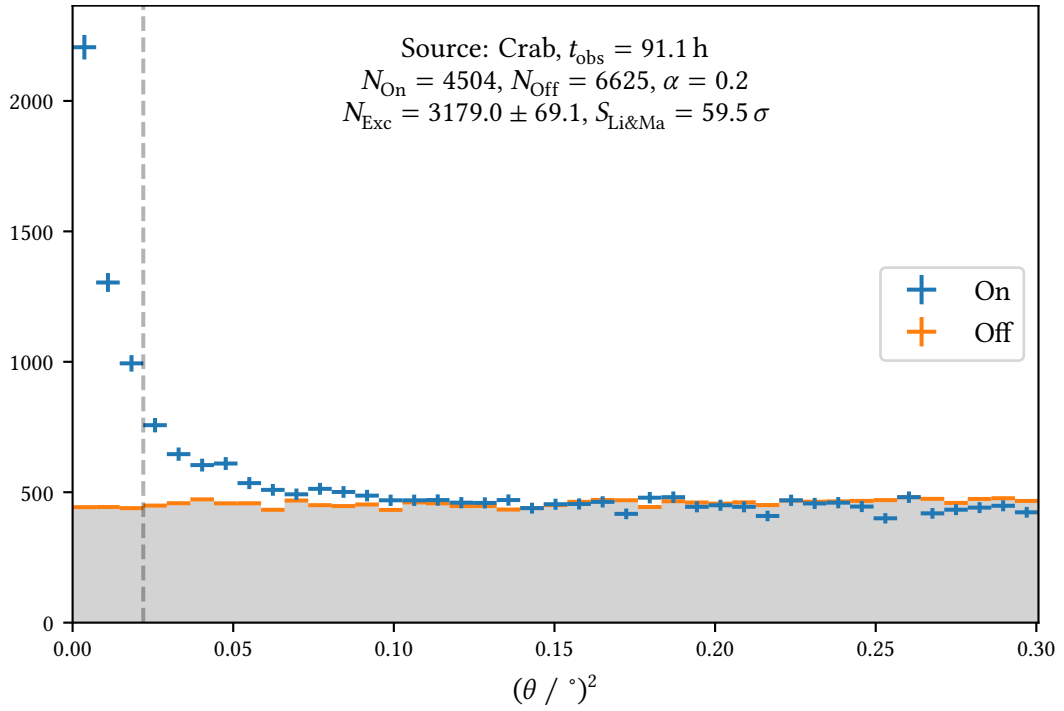


Figure 9: Theta-Squared-Plot for APA = 95 %, see Figure 6.11.

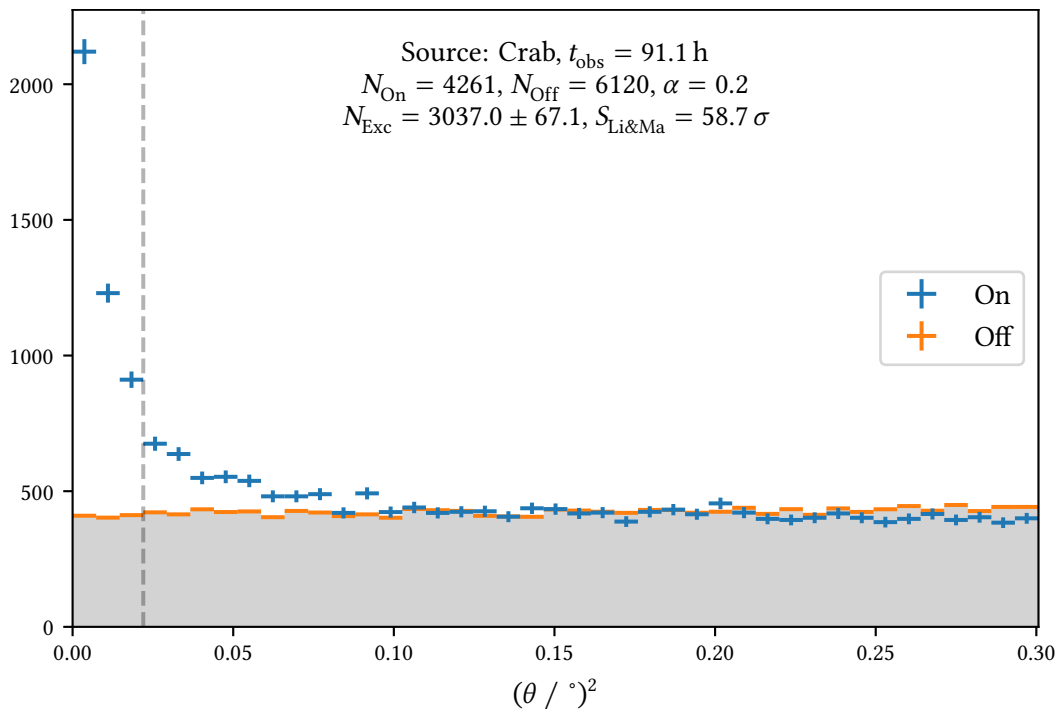


Figure 10: Theta-Squared-Plot for APA = 100 %, see Figure 6.11.

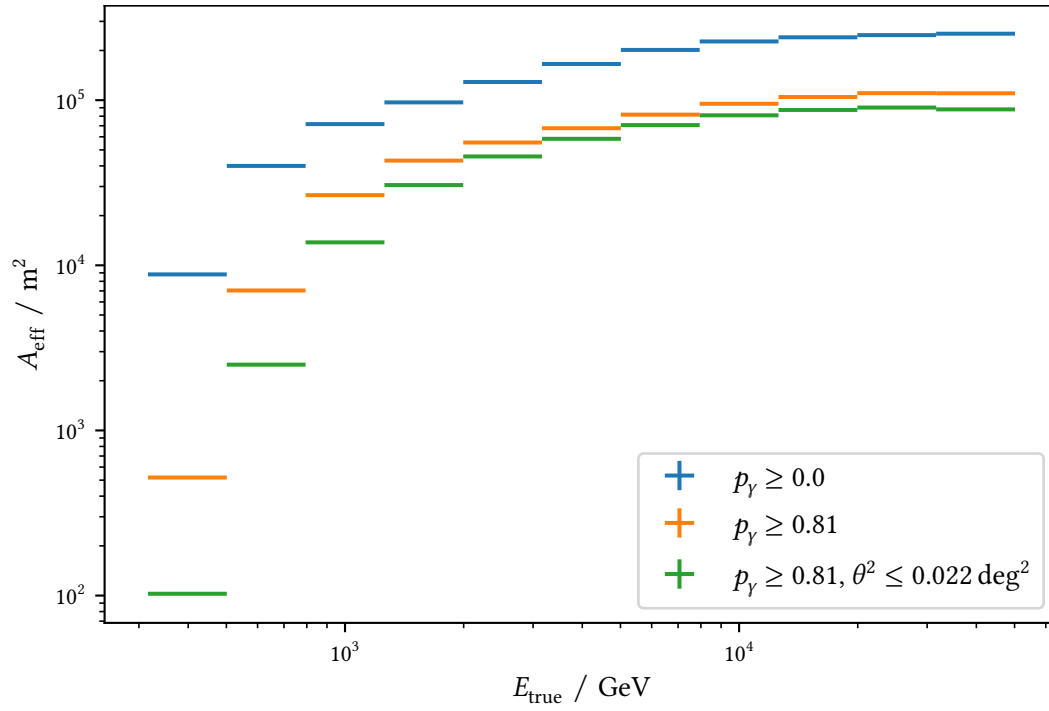


Figure 11: Effective area for APA = 95 %, see Figure 6.17.

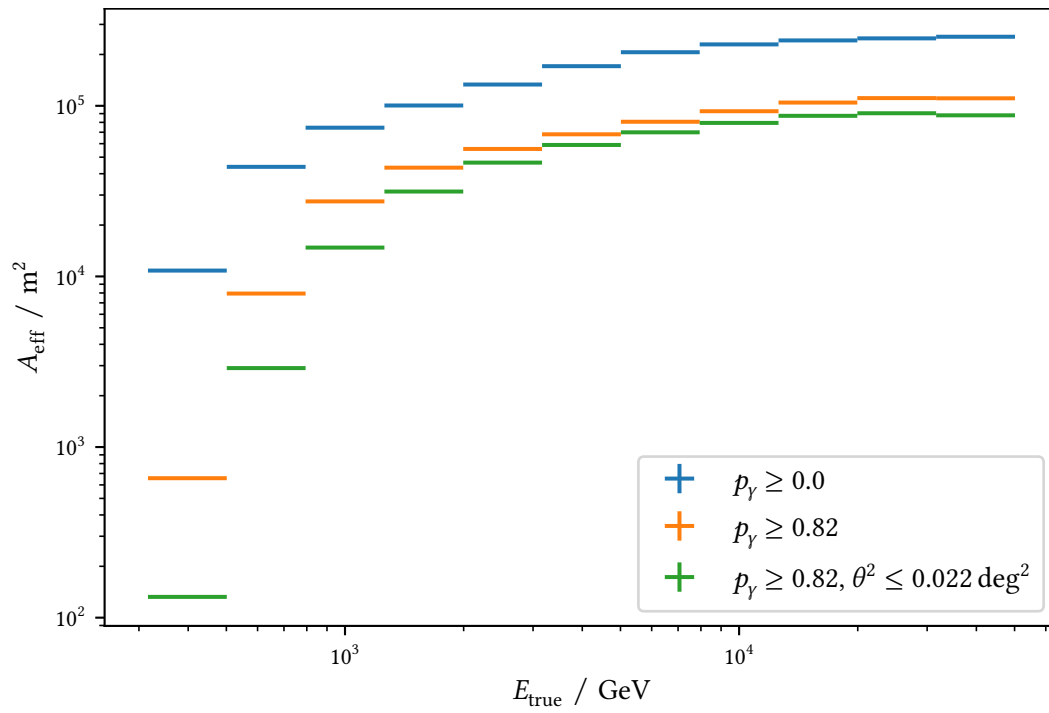


Figure 12: Effective area for APA = 100 %, see Figure 6.17.

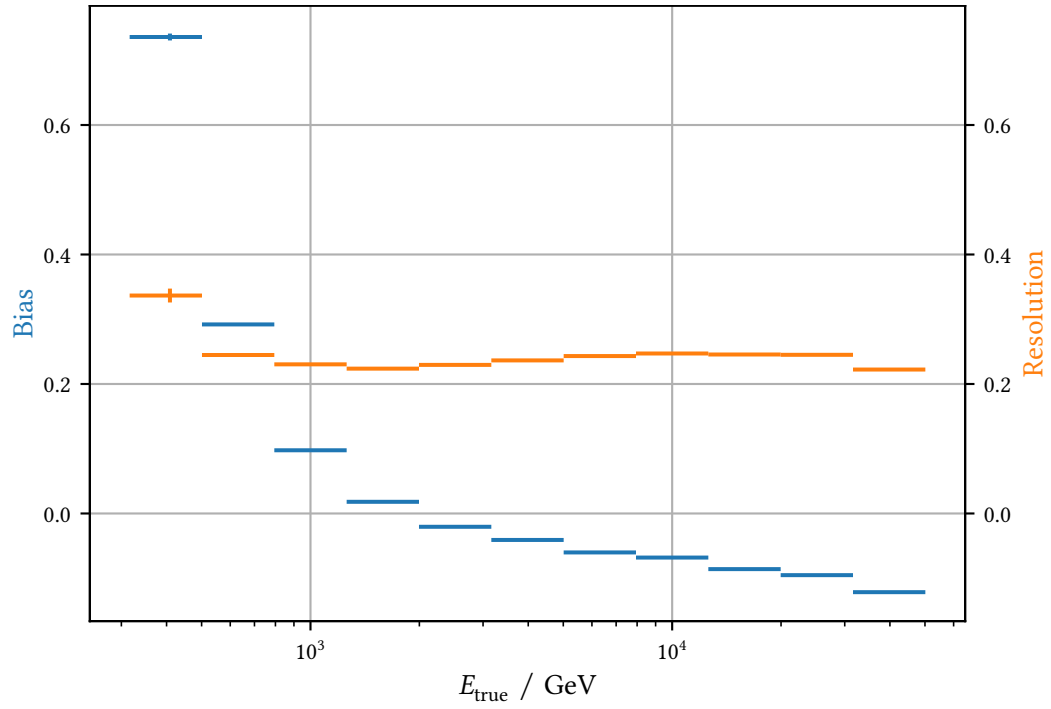


Figure 13: Bias and resolution for APA = 95 %, see Figure 6.21.

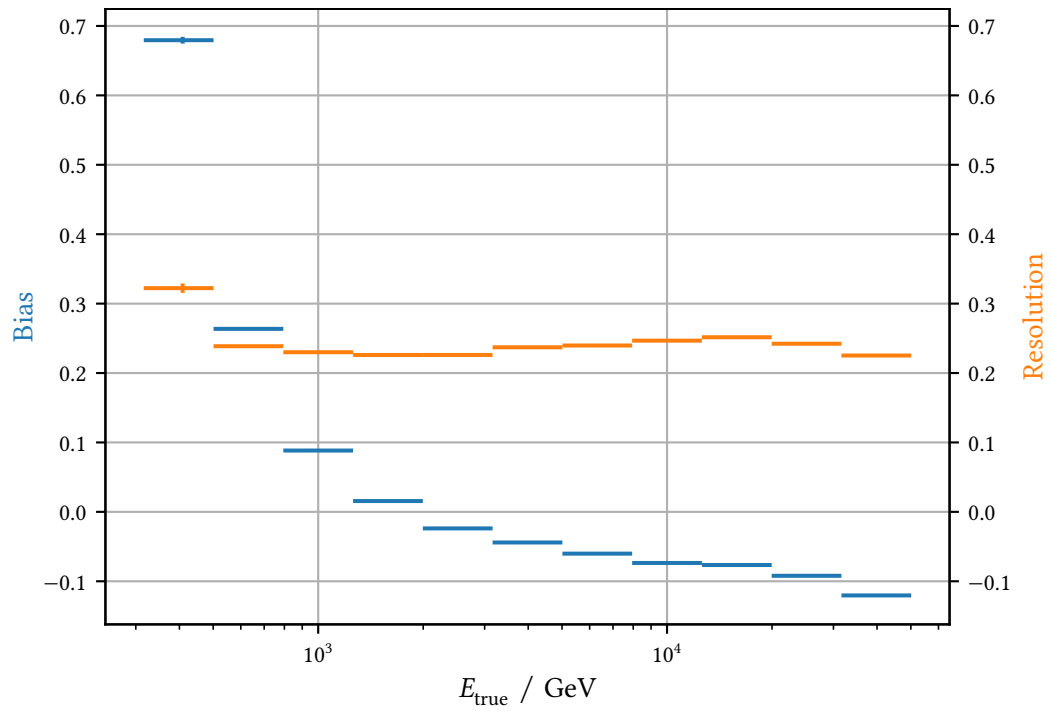


Figure 14: Bias and resolution for APA = 100 %, see Figure 6.21.

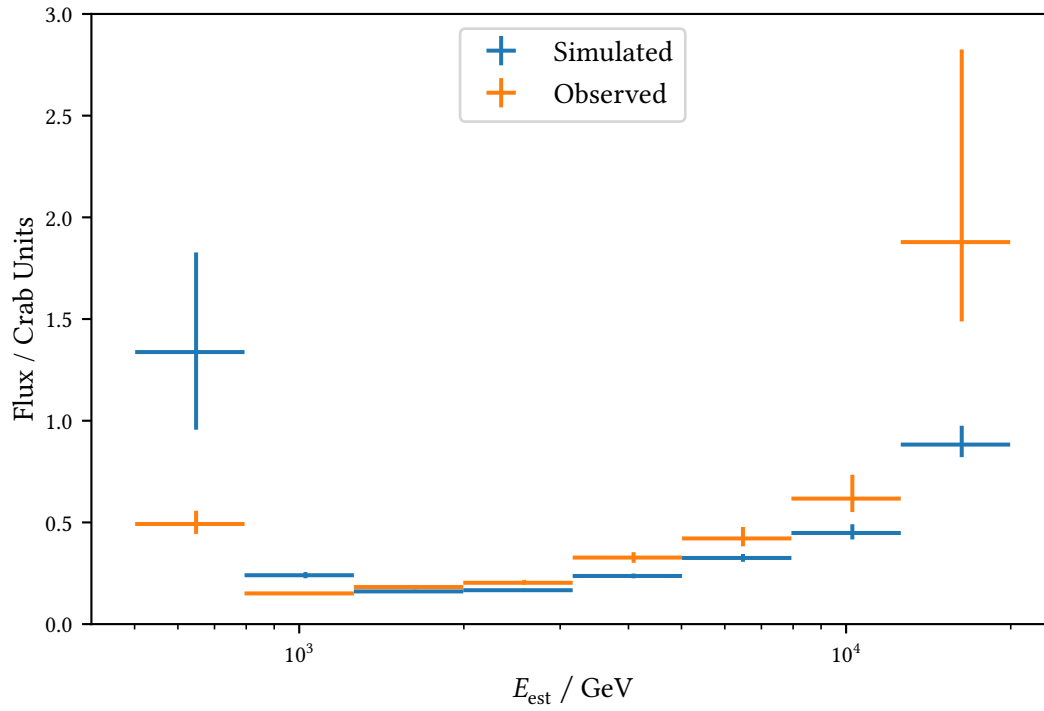


Figure 15: Differential sensitivity for APA = 95 %, see Figure 6.23.

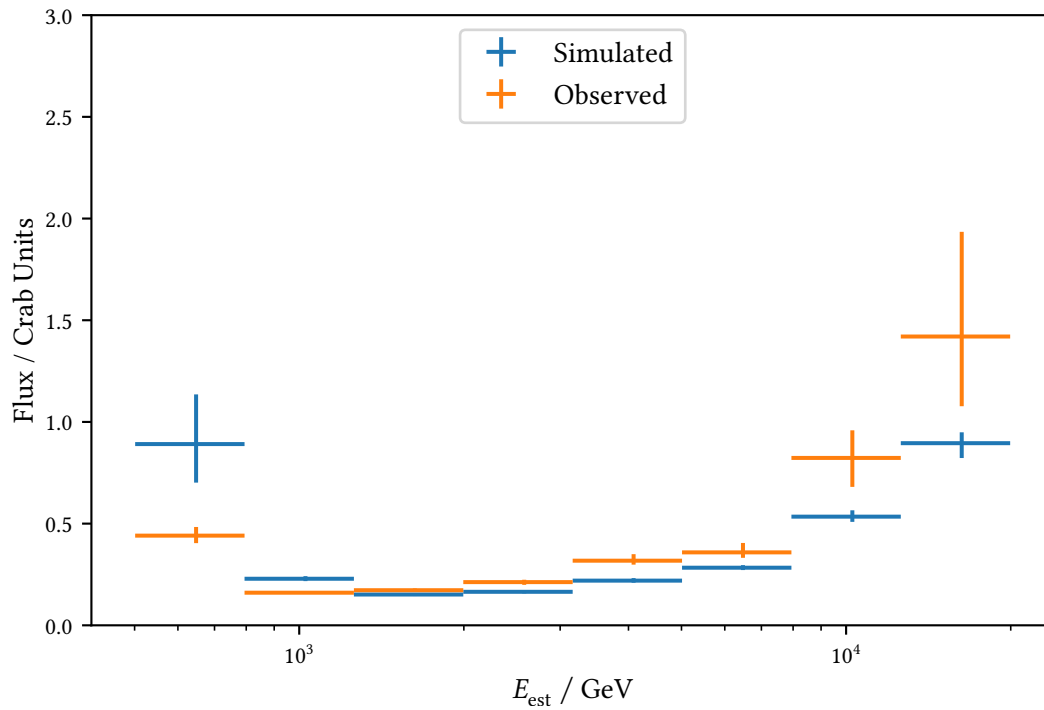


Figure 16: Differential sensitivity for APA = 100 %, see Figure 6.23.

16 Estimated Energy vs. True Energy for Gammas and Protons

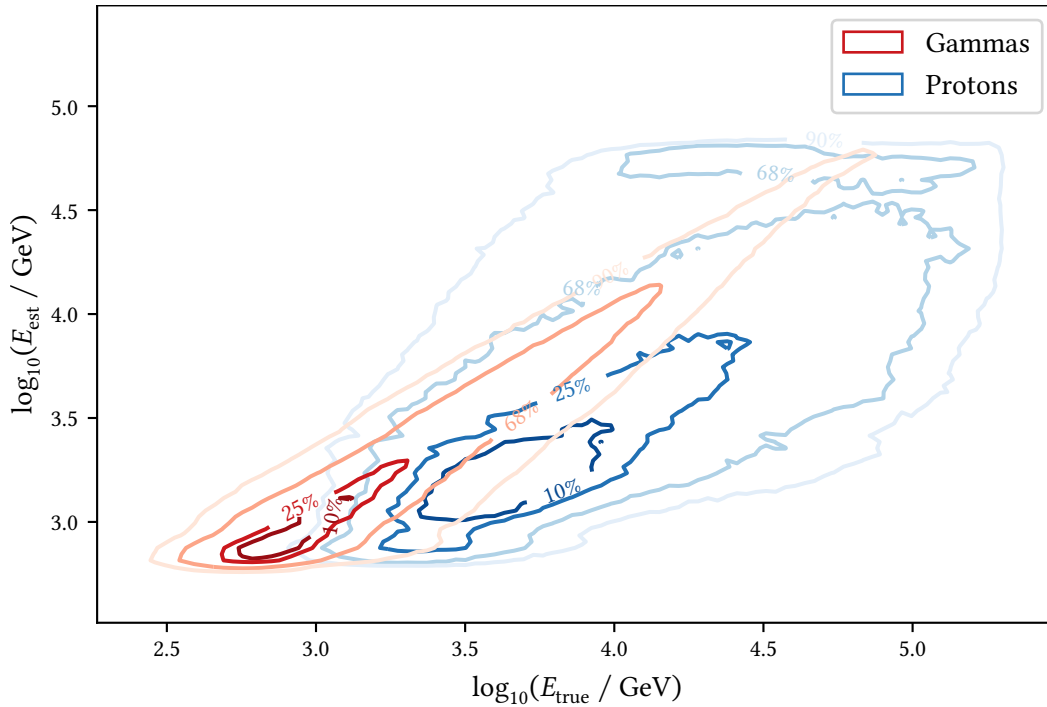


Figure 17: E_{est} vs. E_{true} for simulated proton and gamma ray induced showers. The contours contain the listed percentage of all events.

17 Configuration for the Unfolding

```
# regularization parameter, null = no regularization
# smaller values mean stronger regularization
tau: 0.1

# Estimate background from the off positions
background: True

# reference energy
e_ref:
  value: 1
  unit: GeV

# binning, five bins per decade in energy
e_true_low:
  value: 501.2 # 10^2.7
  unit: GeV
```

```
e_true_high:  
  value: 31622 # 10^4.5  
  unit: GeV  
  
n_bins_true: 9  
  
e_est_low:  
  value: 400  
  unit: GeV  
  
e_est_high:  
  value: 40  
  unit: TeV  
  
n_bins_est: 25  
  
# mcmc paramters  
n_burn_steps: 5000  
n_used_steps: 20000
```


Eidesstattliche Versicherung

Ich versichere hiermit an Eides statt, dass ich die vorliegende Dissertation mit dem Titel „Monitoring the High Energy Universe“ selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt, sowie wörtliche und sinngemäße Zitate kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Ort, Datum

Unterschrift

Belehrung

Wer vorsätzlich gegen eine die Täuschung über Prüfungsleistungen betreffende Regelung einer Hochschulprüfungsordnung verstößt, handelt ordnungswidrig. Die Ordnungswidrigkeit kann mit einer Geldbuße von bis zu 50 000 € geahndet werden. Zuständige Verwaltungsbehörde für die Verfolgung und Ahndung von Ordnungswidrigkeiten ist der Kanzler/die Kanzlerin der Technischen Universität Dortmund. Im Falle eines mehrfachen oder sonstigen schwerwiegenden Täuschungsversuches kann der Prüfling zudem exmatrikuliert werden (§ 63 Abs. 5 Hochschulgesetz –HG–).

Die Abgabe einer falschen Versicherung an Eides statt wird mit Freiheitsstrafe bis zu 3 Jahren oder mit Geldstrafe bestraft.

Die Technische Universität Dortmund wird ggf. elektronische Vergleichswerkzeuge (wie z. B. die Software „turnitin“) zur Überprüfung von Ordnungswidrigkeiten in Prüfungsverfahren nutzen.

Die oben stehende Belehrung habe ich zur Kenntnis genommen.

Ort, Datum

Unterschrift

Acknowledgements

I want to thank Prof. Dr. Dr. Wolfgang Rhode for giving me all the opportunities I could have wished for ever since starting to work on my Bachelor's thesis in 2014. I'm also grateful to all members of this working group for lots of deep discussions and fun hours outside of work. It made my time here really worthwhile.

During my PHD thesis, most of my work was funded by the DFG through the Collaborative Research Center 876 "Providing Information by Resource-Constrained Data Analysis".

I want to especially thank Jens Buß, who basically left me no other choice but to join this working group, supervised my Bachelor's and Master's thesis and was my office mate for most of my time here.

Through all this time, Kai Brügge has been my office mate, pair-programming partner, travel companion and drinking buddy. Our shared interests in photo- and typography, programming and fine beers bridged the gap created by me not supporting the correct football team. Sometimes, people were really surprised, when only one of us came to a meeting or conference. For all this, I am truly grateful.

Andrea Teichmann is the heart and soul of our group, thank you for all the support over the years.

For all the discussions about programming, software development and life in general I want to thank Dominik Neise. I learned a lot with and from you.

For enabling me to present my work at the International Cosmic Rays Conference in Busan, Korea my thanks go to the DAAD.

For discussions about Cherenkov astronomy, open science, physics, analysis techniques, data formats and more I want to thank Konrad Bernlöhr, Christoph Deil, Tarek Hassan, Karl Kosack and Abelardo Moralejo.

To Dieter Heck, thank you for promptly answering CORSIKA support emails although "I have reduced my activity in cosmic rays to half a day per week as I am now more than 14 years in retirement."

I also owe a deep thanks to everyone involved with PeP et al., but especially to Christophe, Tobi, Henning, Marie, Kevin (all of them), Igor and Ismo. Igor, Kevin and Ismo started the Toolbox Workshop, which first introduced me to my programming "mother language" and which I co-organized in the following years. The tools taught in this workshop are the foundation of this thesis and opened many doors for me. These three also first introduced me to the concepts of Open Source Software and reproducible science and made me think it would be a good idea to try Arch as my first Linux distribution.

For proof-reading this document, I want thank Alexander, Kai, Lena and Thorben.

All this would have been impossible without the support of my family. Especially, I want to thank my father for encouraging my curiosity and skepticism and my mother and brother for enduring our dinner table discussions.

Finally, to Lena, thank you for everything.